(12) DEMANDE INTERNATIONALE PUBLIÉE EN VERTU DU TRAITÉ DE COOPÉRATION EN MATIÈRE DE BREVETS (PCT)

(19) Organisation Mondiale de la Propriété Intellectuelle
Bureau international

(43) Date de la publication internationale
12 avril 2001 (12.04.2001)

PCT

(10) Numéro de publication internationale
WO 01/26278 A1

(51) Classification internationale des brevets[7]: H04L 9/32

(21) Numéro de la demande internationale: PCT/FR00/02715

(22) Date de dépôt international: 29 septembre 2000 (29.09.2000)

(25) Langue de dépôt: français

(26) Langue de publication: français

(30) Données relatives à la priorité:

99/12465	1 octobre 1999 (01.10.1999)	FR
99/12467	1 octobre 1999 (01.10.1999)	FR
99/12468	1 octobre 1999 (01.10.1999)	FR
00/09644	21 juillet 2000 (21.07.2000)	FR

(71) Déposants *(pour tous les États désignés sauf US)*: **FRANCE TELECOM** [FR/FR]; 6, place d'Alleray, F-75015 Paris (FR). **TELEDIFFUSION DE FRANCE** [FR/FR]; 10, rue d'Oradour-sur-Glane, F-75732 Paris Cedex 15 (FR). **MATH RIZK** [BE/BE]; Verte Voie, Boîte 5, B-1348 Louvain-la-Neuve (BE).

(72) Inventeurs; et
(75) Inventeurs/Déposants *(pour US seulement)*: **GUILLOU, Louis** [FR/FR]; 16, rue de l'Ise, F-35230 Bourgbarre (FR). **QUISQUATER, Jean-Jacques** [BE/BE]; 3, avenue des Canards, B-1640 Rhode Saint Genese (BE).

(74) Mandataire: **VIDON, Patrice**; Le Nobel, 2, allée Antoine Becquerel, Boîte postale 90333, F-35703 Rennes Cedex 7 (FR).

(81) États désignés *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) États désignés *(régional)*: brevet ARIPO (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), brevet eurasien (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), brevet européen (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), brevet OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Publiée:
— *Avec rapport de recherche internationale.*

En ce qui concerne les codes à deux lettres et autres abréviations, se référer aux "Notes explicatives relatives aux codes et abréviations" figurant au début de chaque numéro ordinaire de la Gazette du PCT.

(54) **Title:** SET OF PARTICULAR KEYS FOR PROVING AUTHENTICITY OF AN ENTITY OR THE INTEGRITY OF A MESSAGE

(54) **Titre:** JEUX DE CLES PARTICULIERS DESTINES A PROUVER L'AUTHENTICITE D'UNE ENTITE OU L'INTEGRITE D'UN MESSAGE

(57) **Abstract:** The invention concerns a set of particular keys designed to prove the authenticity of an entity or the integrity of a message. The proof is established by a set of keys comprising: m (≥1) pairs of private Q_i and public $G_i = g_i^2$ values; a public module n consisting of the product of f(≥2) prime factors; an exponent $v=2^k$ (k > 1), linked by relationships of the type: $G_i . Q_i^v \equiv 1$. mod n or $G_i \equiv Q_i^v$ mod n . The set of keys is produced such that: among the m numbers obtained by increasing Q_i or its inverse modulo n to modulo n square, k-1 times rank, at least one of them is different from g_i; among the 2m equations: $x^2 \equiv g_i$ mod n, $x^2 \equiv -g_i$ mod n at least one of them has solutions in x in the ring of the modulo n integers.

(57) **Abrégé:** La preuve est établie au moyen de jeux de clés comprenant: m (≥ 1) couples de valeurs privées Q_i et publiques $G_i= g_i^2$; un module public n constitué par le produit de f(≥ 2) facteurs premiers un exposant $v=2^k$ (k> > 1), liés par des relations du type: $G_i.Q_i^v= \equiv 1$. mod n ou $G_i= \equiv Q_i^v$ mod n. Les jeux de clés sont produits de telle sorte que: parmi les m nombres obtenus en élevant Q_i ou son inverse modulo n au carré modulo n, k-1 fois de rang, au moins l'un d'entre eux est différent de ± g_i parmi les 2m équations: $x^2 g_i$ mod n, $x^2 = - g_i$ mod n, au moins l'une d'entre elles a des solutions en x dans l'anneau des entiers modulo n.

WO 01/26278 A1

Jeux de clés particuliers destinés à prouver l'authenticité d'une entité ou l'intégrité d'un message.

La présente invention concerne le domaine technique des procédés, des systèmes ainsi que des dispositifs destinés à prouver l'authenticité d'une entité et/ou l'intégrité et/ou l'authenticité d'un message.

Le brevet EP 0 311 470 B1 dont les inventeurs sont Louis Guillou et Jean-Jacques Quisquater décrit un tel procédé. On y fera ci-après référence en le désignant par les termes : "brevet GQ" ou "procédé GQ". Par la suite on désignera parfois par "GQ2", "invention GQ2" ou "technologie GQ2" de nouveaux développements de la technologie GQ faisant l'objet des demandes pendantes déposées le même jour que la présente demande par France Télécom, TDF et la Société Mathrizk et ayant pour inventeur Louis Guillou et Jean-Jacques Quisquater. Les traits caractéristiques de ces demandes pendantes sont rappelés chaque fois que cela est nécessaire dans la description qui suit.

Selon le procédé GQ, une entité appelée " autorité de confiance " attribue une identité à chaque entité appelée " témoin " et en calcule la signature RSA; durant un processus de personnalisation, l'autorité de confiance donne identité et signature au témoin. Par la suite, le témoin proclame : " *Voici mon identité ; j'en connais la signature RSA.* " Le témoin prouve sans la révéler qu'il connaît la signature RSA de son identité. Grâce à la clé publique de vérification RSA distribuée par l'autorité de confiance, une entité appelée " contrôleur " vérifie sans en prendre connaissance que la signature RSA correspond à l'identité proclamée. Les mécanismes utilisant le procédé GQ se déroulent " sans transfert de connaissance ". Selon le procédé GQ, le témoin ne connaît pas la clé privée RSA avec laquelle l'autorité de confiance signe un grand nombre d'identités.

La technologie GQ précédemment décrite fait appel à la technologie RSA. Mais si la technologie RSA dépend bel et bien de la factorisation du

module n, cette dépendance n'est pas une équivalence, loin s'en faut, comme le démontrent les attaques dites "multiplicatives" contre les diverses normes de signature numérique mettant en œuvre la technologie RSA.

L'objectif de la technologie GQ2 est double : d'une part, améliorer les performances par rapport à la technologie RSA ; d'autre part, éviter les problèmes inhérents à la technologie RSA. La connaissance de la clé privée GQ2 est équivalente à la connaissance de la factorisation du module n. Toute attaque au niveau des triplets GQ2 se ramène à la factorisation du module n : il y a cette fois équivalence. Avec la technologie GQ2, la charge de travail est réduite, tant pour l'entité qui signe ou qui s'authentifie que pour celle qui contrôle. Grâce à un meilleur usage du problème de la factorisation, tant en sécurité qu'en performance, la technologie GQ2 évite les inconvénients présentés par la technologie RSA.

Le procédé GQ met en œuvre des calculs modulo des nombres de 512 bits ou davantage. Ces calculs concernent des nombres ayant sensiblement la même taille élevés à des puissances de l'ordre de $2^{16} + 1$. Or les infrastructures microélectroniques existantes, notamment dans le domaine des cartes bancaires, font usage de microprocesseurs auto-programmables monolithiques dépourvus de coprocesseurs arithmétiques. La charge de travail liée aux multiples opérations arithmétiques impliquées par des procédés tels que le procédé GQ, entraîne des temps de calcul qui dans certains cas s'avèrent pénalisant pour les consommateurs utilisant des cartes bancaires pour acquitter leurs achats. Il est rappelé ici, qu'en cherchant à accroître la sécurité des cartes de paiement, les autorités bancaires posent un problème particulièrement délicat à résoudre. En effet, il faut traiter deux questions apparemment contradictoires : augmenter la sécurité en utilisant des clés de plus en plus longues et distinctes pour chaque carte tout en évitant que la charge de travail n'entraîne des temps

de calcul prohibitifs pour les utilisateurs. Ce problème prend un relief particulier dans la mesure où, en outre, il convient de tenir compte de l'infrastructure en place et des composants microprocesseurs existants.

La technologie GQ2 apporte une solution à ce problème tout en renforçant la sécurité.

La technologie GQ2 met en œuvre des facteurs premiers ayant des propriétés particulières. Différentes techniques existent pour produire ces facteurs premiers. La présente invention a pour objet un procédé permettant de produire de manière systématique de tels facteurs premiers. Elle concerne aussi l'application qui peut être faite de ceux-ci plus particulièrement dans la mise en œuvre de la technologie GQ2. On souligne dès à présent que ces facteurs premiers particuliers et le procédé permettant de les obtenir sont susceptibles d'application en dehors du champ de la technologie GQ2.

L'invention s'applique à un procédé destiné à prouver à une entité contrôleur,

- l'authenticité d'une entité et/ou
- l'intégrité d'un message **M** associé à cette entité.

Un tel procédé met en œuvre :

- un module public **n** constitué par le produit de **f** facteurs premiers $\mathbf{p_1}$, $\mathbf{p_2}$, ... $\mathbf{p_f}$ (**f** étant supérieur ou égal à 2) ou mettant en œuvre les **f** facteurs premiers,
- **m** nombres de base entiers $\mathbf{g_1}$, $\mathbf{g_2}$, ... $\mathbf{g_m}$ distincts (**m** étant supérieur ou égal à 1), $\mathbf{g_i}$ étant inférieur aux **f** facteurs premiers $\mathbf{p_1}$, $\mathbf{p_2}$, ... $\mathbf{p_f}$,
- **m** couples de valeurs privées $\mathbf{Q_1}$, $\mathbf{Q_2}$, ... $\mathbf{Q_m}$ et publiques $\mathbf{G_1}$, $\mathbf{G_2}$, ... $\mathbf{G_m}$ (**m** étant supérieur ou égal à 1) ou des paramètres dérivés de ceux-ci.

Ledit module et lesdites valeurs privées et publiques sont liés par des relations du type :

$$\mathbf{G_i \cdot Q_i^v \equiv 1 \cdot \bmod n \text{ ou } G_i \equiv Q_i^v \bmod n}$$

ladite valeur publique $\mathbf{G_i}$ étant le carré $\mathbf{g_i^2}$ du nombre de base,
v désignant un exposant public de la forme :

$$\mathbf{v = 2^k}$$

où **k** est un paramètre de sécurité plus grand que 1.

Le procédé selon l'invention comprend l'étape de produire les **f** facteurs premiers $\mathbf{p_{1,}}, \mathbf{p_{2,}}, \ldots \mathbf{p_f}$ et/ou les **m** nombres de base $\mathbf{g_{1,}}, \mathbf{g_{2,}}, \ldots \mathbf{g_m}$ de telle sorte que les conditions suivantes soient satisfaites.

Première condition :

Selon la première condition, chacune des équations :

$$\mathbf{x^v \equiv g_i^2 \bmod n} \qquad (1)$$

a des solutions en x dans l'anneau des entiers modulo **n.**

Deuxième condition :

Selon la deuxième condition, dans le cas où $\mathbf{G_i \equiv Q_i^v \bmod n}$, parmi les **m** nombres $\mathbf{q_i}$ obtenus en élevant $\mathbf{Q_i}$ au carré modulo n, k-1 fois de rang, l'un d'entre eux est différent de $\pm\, \mathbf{g_i}$ (c'est-à-dire est non trivial).

Selon la deuxième condition, dans le cas où $\mathbf{G_i \,.\, Q_i^v \equiv 1 \bmod n}$, parmi les **m** nombres $\mathbf{q_i}$ obtenus en élevant l'inverse de $\mathbf{Q_i}$ modulo n au carré modulo n, k-1 fois de rang, l'un d'entre eux est différent de $\pm\, \mathbf{g_i}$ (c'est-à-dire est non trivial).

Il est précisé ici que selon une notation courante $\pm\, \mathbf{g_i}$ représente les nombres $\mathbf{g_i}$ et $\mathbf{n\text{-}g_i}$.

Troisième condition :

Selon la troisième condition, parmi les **2m** équations :

$$\mathbf{x^2 \equiv g_i \bmod n} \qquad (2)$$

$$\mathbf{x^2 \equiv - g_i \bmod n} \qquad (3)$$

au moins l'une d'entre elles a des solutions en x dans l'anneau des entiers modulo **n.**

Le procédé selon l'invention pour produire les **f** facteurs premiers $\mathbf{p_{1,}}, \mathbf{p_{2,}}, \ldots \mathbf{p_f}$, et/ou les **m** nombres de base $\mathbf{g_{1,}}, \mathbf{g_{2,}}, \ldots \mathbf{g_m}$ comprend l'étape de choisir

en premier :

• le paramètre de sécurité **k**,

• les **m** nombres de base $\mathbf{g_{1,}}$, $\mathbf{g_{2,}}$, ... $\mathbf{g_m}$, et/ou les **f** facteurs premiers $\mathbf{p_{1,}}$, $\mathbf{p_{2,}}$, ... $\mathbf{p_f}$, selon qu'il s'agit respectivement de produire les **f** facteurs premiers $\mathbf{p_{1,}}$, $\mathbf{p_{2,}}$, ... $\mathbf{p_f}$ ou les **m** nombres de base $\mathbf{g_{1,}}$, $\mathbf{g_{2,}}$, ... $\mathbf{g_m}$.

De préférence, les **m** nombres de base $\mathbf{g_1}$, $\mathbf{g_2}$, ... $\mathbf{g_m}$, sont choisis au moins en partie parmi les premiers nombres entiers.

De préférence, le paramètre de sécurité **k** est un petit nombre entier, notamment inférieur à 100.

De préférence, la taille du module **n** est supérieure à plusieurs centaines de bits.

De préférence, les **f** facteurs premiers $\mathbf{p_{1,}}$, $\mathbf{p_{2,}}$, ... $\mathbf{p_f}$ ont une taille voisine de la taille du module **n** divisé par le nombre **f** de facteurs.

Pour tester la première condition, on vérifie la compatibilité des nombres **k**, **p**, **g** en mettant en œuvre l'algorithme ci-après, où **h** désigne un nombre tel que $\mathbf{2^h}$ divise le rang de **g** par rapport à **p** et tel que $\mathbf{2^{h+1}}$ ne le divise pas. On calcule **h** à partir du symbole de Legendre $\mathbf{(g \mid p)}$ et d'un nombre **b** égal à une racine $\mathbf{2^t}$ - ième primitive de l'unité dans **CG(p),** où le symbole de Legendre $(g_i \mid p_j)$ et t ont le sens défini ci-après dans la description.

Voici les étapes de cet algorithme :

• si $(g \mid p) = -1$ alors $h = t$,

• si $(g \mid p) = +1$ avec $t = 1$, alors $h = 0$,

• si $(g \mid p) = +1$ avec $t > 1$, on procède comme indiqué ci-après.

On applique la clé $\langle(p-1+2^t)/ 2^{t+1}, p\rangle$ à G, on obtient ainsi un résultat w :

• si $w = +g$, alors $h = 0$,

• si $w = p-g$, alors $h = +1$.

Si w est différent de +g ou de p-g (dans ce cas t est supérieur à 2), on met en œuvre un sous-module de calcul. On initialise la variable c en lui attribuant la valeur b, puis on itére les étapes suivantes du sous-module de

calcul pour des valeurs de i allant de t-1 à 2 :

étape 1 : on applique la clé $\langle 2^i, p\rangle$ à w/g(modp),

* si le résultat obtenu est égal à +1, on passe à l'étape 2,

* si le résultat obtenu est égal à -1, on attribue à h la valeur i et on remplace w par w.c(modp),

étape 2 : on remplace c par c^2 (modp).

La valeur de h recherchée est celle obtenue la dernière fois où l'application de la clé $\langle 2^i, p\rangle$, conformément à l'étape 1, a produit un résultat égal à –1.

On rappelle que :

- k, g, p sont incompatibles lorsque h>1 et lorsque k+h >t+1,

- k, g, p sont compatibles lorsque h = 0 ou 1, quelle que soit la valeur de k, ou lorsque h>1 et lorsque k+h ≤ t+1.

Pour tester la deuxième condition, on vérifie qu'au moins un jeu $\{\delta_{i,1} \ldots \delta_{i,f}\}$ est variable ou nul, (δ à le sens défini ci-après dans la description).

Pour tester la troisième condition, on vérifie qu'il existe un nombre de base g_i de g_1 à g_m tel que les f symboles de Legendre $(g_i \mid p_1)$ à $(g_i \mid p_f)$ soient tous égaux à +1 ou bien les f symboles de Legendre $(-g_i \mid p_1)$ à $(-g_i \mid p_f)$ soient tous égaux à +1.

Pour calculer les **f.m** composantes privées $\mathbf{Q_{i,j}}$ des valeurs privées $\mathbf{Q_1, Q_2, \ldots Q_m}$ ($\mathbf{Q_{i,j} \equiv Q_i \bmod p_j}$), dans le cas où $\mathbf{G_i \equiv Q_i^v \bmod n}$, on procède comme suit, en distinguant les cas selon les valeurs de t.

Cas où t = 1 (c'est-à-dire si $p_j \equiv 3$ (mod4)).

• on calcule un nombre s_j tel que $s_j \equiv ((p_j + 1)/4)^k\ (\mathrm{mod}(p_j - 1)/2)$,

• on en déduit la clé $\langle s_j, p_j\rangle$,

• on applique la clé $\langle s_j, p_j\rangle$ à $\mathbf{G_i}$,

• on obtient ainsi : $w \equiv \mathbf{G_i}^{sj}$ (mod p_j).

Les deux valeurs possibles de $\mathbf{Q_{i,j}}$ sont w, p_j –w.

Cas où t = 2 (c'est-à-dire si $p_j \equiv 5$ (mod8)) :

• on calcule un nombre s_j tel que $s_j \equiv ((p_j + 3)/8)^k\ (\mathrm{mod}(p_j - 1)/4)$,

• on en déduit la clé $\langle s_j, p_j \rangle$,

• on applique la clé $\langle s_j, p_j \rangle$ à $\mathbf{G_i}$,

• on obtient ainsi : $w \equiv \mathbf{G_i}^{sj}$ (mod p_j) et $w' \equiv w.z$(mod p_j),

où z a le sens défini ci-après dans la description.

Les quatre valeurs possibles de $\mathbf{Q_{i,j}}$ sont w, p_j –w, w', p_j –w'.

Cas où t >2 (c'est-à-dire si $p_j \equiv 2^t+1$ (mod2^{t+1})) avec h=0 ou avec h = 1,

• on calcule s_j tel que $s_j \equiv ((p_j -1 + 2^t)/ 2^{t+1})^k$ (mod(p_j –1)/ 2^t),

• on en déduit la clé $\langle s_j, p_j \rangle$,

• on applique la clé $\langle s_j, p_j \rangle$ à $\mathbf{G_i}$,

• on obtient ainsi : $w \equiv \mathbf{G_i}^{sj}$ (mod p_j).

Les $2^{\min(k,t)}$ valeurs possibles de $\mathbf{Q_{i,j}}$ sont égales au produit de w par l'une quelconque des racines $2^{\min(k,t)}$-ièmes de l'unité dans CG(p_j).

Cas où t >2 (c'est-à-dire si $p_j \equiv 2^t+1$ (mod2^{t+1})) avec h>1 et avec $h+k \leq t+1$,

• on calcule s_j tel que $s_j \equiv ((p_j -1 + 2^t)/ 2^{t+1})^{k+h-1}$(mod($p_j$ –1)/ 2^t),

• on en déduit la clé $\langle s_j, p_j \rangle$,

• on applique la clé $\langle s_j, p_j \rangle$ à la puissance 2^{h-1} –ième de $\mathbf{G_i}$,

• on obtient ainsi w,

Les 2^k valeurs possibles de $\mathbf{Q_{i,j}}$ appartiennent à l'ensemble des produits de w par les racines 2^{k+h-1} –ièmes primitives de l'unité dans CG(p_j).

Pour calculer les composantes privées $\mathbf{Q_{i,j}}$ dans le cas où $\mathbf{G_i . Q_i^v \equiv 1 . mod\ n}$, on substitue s_j par $((p_j -1)/2^t) - s_j$ dans la clé $\langle s_j, p_j \rangle$.

L'invention concerne également un procédé faisant application de la méthode permettant de produire les **f** facteurs premiers $\mathbf{p_1, p_2, \ldots p_f}$ ou les **m** nombres de base $\mathbf{g_1, g_2, \ldots g_m}$.

Ledit procédé est destiné à prouver à une entité contrôleur

- l'authenticité d'une entité et/ou

- l'intégrité d'un message **M** associé à cette entité,

au moyen de **m** couples de valeurs privées $\mathbf{Q_1, Q_2, \ldots Q_m}$ et publiques $\mathbf{G_1, G_2, \ldots G_m}$ (**m** étant supérieur ou égal à 1) ou des paramètres dérivés de

ceux-ci, notamment au moyen des composantes privées $\mathbf{Q_{i,j}}$.

Ledit procédé met en œuvre selon les étapes ci-après une entité appelée témoin.

Ladite entité témoin dispose des **f** facteurs premiers $\mathbf{p_i}$ et/ou des paramètres des restes chinois des facteurs premiers et/ou du module public **n** et/ou des **m** valeurs privées $\mathbf{Q_i}$ et/ou des **f.m** composantes privées $\mathbf{Q_{i,j}}$ des valeurs privées $\mathbf{Q_i}$ et de l'exposant public.

Le témoin calcule des engagements **R** dans l'anneau des entiers modulo **n.** Chaque engagement est calculé :

• soit en effectuant des opérations du type

$$\mathbf{R \equiv r^v \bmod n}$$

où **r** est un aléa tel que $\mathbf{0 < r < n}$,

• soit en appliquant la méthode des restes chinois après avoir effectué des opérations du type

$$\mathbf{R_i \equiv r_i^{\,v} \bmod p_i}$$

où $\mathbf{r_i}$ est un aléa associé au nombre premier $\mathbf{p_i}$ tel que $\mathbf{0 < r_i < p_i}$, chaque $\mathbf{r_i}$ appartenant à une collection d'aléas $\mathbf{\{r_1 , r_2 , \dots r_f\}}$.

Le témoin reçoit un ou plusieurs défis **d**. Chaque défi **d** comportant **m** entiers $\mathbf{d_i}$ ci-après appelés défis élémentaires. Le témoin calcule à partir de chaque défi **d** une réponse **D**,

• soit en effectuant des opérations du type :

$$\mathbf{D \equiv r \,.\, Q_1^{\;d1} \,.\, Q_2^{\;d2} \,.\, \dots\, Q_m^{\;dm} \bmod n}$$

• soit en appliquant la méthode des restes chinois après avoir effectué des opérations du type :

$$\mathbf{D_i \equiv r_i \,.\, Q_{i,1}^{\;d1} \,.\, Q_{i,2}^{\;d2} \,.\, \dots\, Q_{i,m}^{\;dm} \bmod p_i}$$

Ledit procédé est tel qu'il y a autant de réponses **D** que de défis **d** et d'engagements **R.** Chaque groupe de nombres **R, d, D** constitue un triplet noté **{R, d, D}.**

Description

Les objectifs des schémas GQ sont l'authentification dynamique d'entités et de messages ainsi que la signature numérique de messages. Ce sont des schémas « sans transfert de connaissance ». Une entité prouve : elle connaît un ou plusieurs nombres privés. Une autre entité contrôle : elle connaît le ou les nombres publics correspondants. L'entité qui prouve veut convaincre l'entité qui contrôle sans révéler le ou les nombres privés, de façon à pouvoir les utiliser autant de fois que de besoin.

Chaque schéma GQ repose sur un module public composé de grands nombres premiers secrets. Un exposant public v et un module public n forment ensemble une clé de vérification $\langle v, n\rangle$ signifiant « élever à la puissance v modulo n » et mise en œuvre au moyen d'une ou plusieurs équations génériques, toutes du même type, direct : $G \equiv Q^v \pmod n$ ou inverse : $G \times Q^v \equiv 1 \pmod n$. Le type a un effet sur le déroulement des calculs au sein de l'entité qui contrôle, pas au sein de l'entité qui prouve ; en fait, les analyses de sécurité confondent les deux types. Chaque équation générique lie un nombre public G et un nombre privé Q formant ensemble un couple de nombres $\{G, Q\}$. En résumé, chaque schéma GQ met en œuvre un ou plusieurs couples de nombres $\{G, Q\}$ pour la même clé $\langle v, n\rangle$.

Une version classique de schémas GQ, appelée ici GQ1, fait appel à un schéma RSA de signature numérique. La clé de vérification $\langle v, n\rangle$ est alors une clé publique RSA où l'exposant v impair est de préférence un nombre premier. Chaque schéma GQ1 utilise en général un seul couple de nombres $\{G, Q\}$: le nombre public G est déduit de données d'identification selon un mécanisme de format qui fait partie intégrante du schéma RSA de signature numérique. Le nombre privé Q ou bien son inverse modulo n est une signature RSA des données d'identification. L'entité qui prouve démontre la connaissance d'une signature RSA de ses propres données d'identification et cette preuve ne révèle pas la signature qui reste donc secrète pour être

utilisée autant de fois que de besoin.

Les schémas GQ1 mettent généralement en œuvre deux niveaux de clés : la clé privée de signature RSA est réservée à une autorité accréditant des entités se distinguant les unes des autres par des données d'identification. On dit qu'un tel schéma est « basé sur l'identité ». Ainsi, un émetteur de cartes à puce utilise sa clé privée RSA à l'émission de chaque carte pour calculer un nombre privé Q qu'il inscrit comme clé privée diversifiée dans la carte ; ou encore, un client sur un réseau d'ordinateurs utilise sa clé privée RSA à chaque entrée en session pour calculer un nombre privé Q qui sera la clé privée éphémère du client durant la session. Les entités qui prouvent, cartes à puce ou clients en session, connaissent une signature RSA de leurs données d'identification ; elles ne connaissent pas la clé privée RSA qui, dans la hiérarchie des clés, se trouve au niveau immédiatement supérieur.

Cependant, une authentification dynamique d'entités par GQ1 avec un module de 768 bits au niveau d'une autorité demande à peu près la même charge de travail qu'une authentification dynamique d'entités par RSA avec un module de 512 bits à trois facteurs premiers au niveau de chaque entité, ce qui permet à l'entité qui prouve d'utiliser la technique des restes chinois en calculant un résultat modulo chacun des facteurs premiers avant de calculer un résultat modulo leur produit.

Toutefois, la hiérarchie de clés entre une autorité et les entités accréditées n'est pas obligatoire. On peut utiliser GQ1 avec un module propre à l'entité qui prouve, ce qui permet d'utiliser la technique des restes chinois pour réduire les charges de travail de l'entité qui prouve, ce qui ne change pas fondamentalement la charge de travail de l'entité qui contrôle, mis à part le fait qu'un module au niveau de l'entité qui prouve peut être plus court qu'un module au niveau de l'autorité, par exemple 512 bits comparés à 768 bits.

Lorsque l'entité connaît les facteurs premiers de son propre module, pourquoi faire appel à un schéma RSA de signature numérique ??

Une autre version de schémas GQ. appelée ici GQ2 élémentaire, fait appel directement au problème de la factorisation d'un module n. Dans ce contexte, « directement » signifie « sans faire appel à la signature RSA ». L'objectif de GQ2 est bien de réduire les charges de travail, non seulement de l'entité qui prouve mais aussi de l'entité qui contrôle. L'entité qui prouve démontre la connaissance d'une décomposition de son propre module et cette preuve ne révèle pas la décomposition qui reste donc secrète pour être utilisée autant de fois que de besoin. La sécurité du protocole GQ2 est équivalente à la factorisation du module.

Chaque entité qui prouve dispose de son propre module n. Chaque schéma GQ2 met en œuvre un paramètre k, petit nombre plus grand que 1 fixant un exposant public $v = 2^k$, et un ou plusieurs couples de nombres $\{G_1, Q_1\}$ à $\{G_m, Q_m\}$. Chaque nombre public G_i est le carré d'un petit nombre g_i plus grand que 1 et appelé « nombre de base ». Toutes les entités qui prouvent peuvent utiliser le ou les mêmes nombres publics G_1 à G_m. La factorisation du module n et le ou les nombres privés Q_1 à Q_m sont alors au même niveau dans la hiérarchie des clés. Chaque jeu de clés GQ2 élémentaires est défini par deux conditions nécessaires et suffisantes.

- Pour chaque nombre de base, aucune des deux équations $x^2 \equiv \pm g_i \pmod{n}$ n'a de solution en x dans l'anneau des entiers modulo n, c'est-à-dire que les nombres $\pm g_i$ sont deux résidus non quadratiques modulo n.
- Pour chaque nombre de base, l'équation $x^v \equiv g_i^2 \pmod{n}$ où $v = 2^k$ a des solutions en x dans l'anneau des entiers modulo n. Le nombre privé Q_i ou son inverse modulo n est n'importe laquelle de ces solutions.

Compte tenu de la deuxième condition, pour que les nombres $\pm g_i$ soient deux résidus non quadratiques modulo n, le module n doit comporter au moins deux facteurs premiers congrus à 3 (mod 4) par rapport auxquels le symbole de Legendre de g_i diffère. Par conséquent, tout module composé de facteurs premiers dont aucun ou un seul est congru à 3 (mod 4) ne permet

pas d'établir un jeu de clés GQ2 élémentaires, ce qui privilégie les facteurs premiers congrus à 3 (mod 4). Or en prenant au hasard des grands nombres premiers, il s'avère qu'ils sont environ pour moitié congrus à 3 (mod 4) et pour moitié à 1 (mod 4). De ce fait, beaucoup de modules RSA en usage ne permettent pas d'établir des jeux de clés GQ2 élémentaires.

Nous introduisons ici les jeux de clés GQ2 généralisées pour surmonter cette limitation afin de pouvoir utiliser des techniques GQ2 avec n'importe quel module, en particulier, n'importe quel module RSA ; ils reposent sur deux principes nécessaires et suffisants.

Le premier principe reproduit la deuxième condition de GQ2 élémentaire.

— Pour chaque nombre de base g_1 à g_m, l'équation $x^v \equiv g_i^2 \pmod{n}$ où $v = 2^k$ a des solutions en x dans l'anneau des entiers modulo n.

Parce que le nombre privé Q_i ou bien son inverse modulo n est une solution à l'équation, $k-1$ carrés successifs modulo n le transforment en un nombre q_i qui est une racine carrée de G_i dans l'anneau des entiers modulo n. Selon que le nombre q_i est égal à l'un des deux nombres g_i ou $n-g_i$, ou différent des deux nombres g_i et $n-g_i$, nous disons qu'il est trivial ou non. Lorsqu'un nombre q_i est non trivial, n qui divise $q_i^2-g_i^2$ ne divise ni q_i-g_i ni q_i+g_i. Tout nombre q_i non trivial révèle donc une décomposition du module n.

$$n = \text{pgcd}(n, q_i-g_i)\times\text{pgcd}(n, q_i+g_i)$$

Le deuxième principe élargit la première condition de GQ2 élémentaire.

— Parmi les nombres q_1 à q_m, au moins un nombre q_i est non trivial.

Observons que si un nombre q_i existe alors que les nombres $\pm g_i$ sont deux résidus non quadratiques dans l'anneau des entiers modulo n, le nombre q_i est manifestement non trivial. Ainsi, les jeux de clés GQ2 élémentaires font bien partie des jeux de clés GQ2 généralisées qui permettent d'utiliser n'importe quel module, c'est-à-dire toute composition de grands nombres premiers congrus indifféremment à 3 ou à 1 (mod 4) dont au moins deux sont distincts. Par contre, beaucoup de jeux de clés GQ2 généralisées ne

sont pas des jeux de clés GQ2 élémentaires. Chaque jeu de clés GQ2 généralisées est dans l'un des deux cas suivants.

- Lorsque les $2\times m$ nombres $\pm g_1$ à $\pm g_m$ sont tous des résidus non quadratiques, c'est un jeu de clés GQ2 élémentaires.
- Lorsque parmi les $2\times m$ nombres $\pm g_1$ à $\pm g_m$, il y a au moins un résidu quadratique, ce n'est pas un jeu de clés GQ2 élémentaires; c'est ce que nous appelons ici un jeu de clés GQ2 complémentaires.

La présente invention porte sur les jeux de clés GQ2 complémentaires, par définition, ces jeux de clés GQ2 généralisées qui ne sont pas élémentaires. Outre les deux principes précédents, un tel jeu doit satisfaire un troisième principe.

— Parmi les $2\times m$ nombres $\pm g_1$ à $\pm g_m$, il y a au moins un résidu quadratique.

Pour appréhender le problème et comprendre la solution que nous en donnons, c'est-à-dire l'invention, analysons d'abord la décomposition du module n révélée par un nombre q non trivial, puis rappelons la technique des restes chinois, puis, la notion de rang dans un corps de Galois $CG(p)$; puis, étudions les fonctions « élever au carré » dans $CG(p)$ et « prendre une racine carrée » d'un résidu quadratique dans $CG(p)$; enfin, analysons l'applicabilité des trois principes énoncés ci-dessus.

Analyse des décompositions du module — De même que le module n se décompose en f facteurs premiers p_1 à p_f, l'anneau des entiers modulo n se décompose en f corps de Galois $CG(p_1)$ à $CG(p_f)$. Dans chaque corps, il y a deux racines carrées de l'unité, à savoir ± 1. Dans l'anneau, il y a donc 2^f racines carrées de l'unité. Chaque nombre privé Q_1 à Q_m définit un nombre $\Delta_i = q_i / g_i \pmod n$ qui est une de ces 2^f racines carrées de l'unité dans l'anneau ; en d'autres termes, n divise Δ_i^2-1.

- Lorsque q_i est trivial, c'est-à-dire $\Delta_i = \pm 1$, n divise Δ_i-1 ou bien Δ_i+1 et donc Δ_i ne révèle pas de décomposition du module n.
- Lorsque q_i est non trivial, c'est-à-dire $\Delta_i \neq \pm 1$, n ne divise ni Δ_i-1 ni Δ_i+1

et donc Δ_i révèle une décomposition, $n = \text{pgcd}(n, \Delta_i-1)\times\text{pgcd}(n, \Delta_i+1)$, résultant de la valeur de Δ_i dans chaque corps : le ou les facteurs premiers divisant Δ_i-1 d'un côté, celui ou ceux divisant Δ_i+1 de l'autre.

Examinons les règles de composition multiplicative des nombres q. Deux nombres $\{q_1, q_2\}$ donnent un nombre composé $q_1\times q_2 \pmod n$.

- Lorsque q_1 est non trivial et q_2 trivial, le nombre composé $q_1\times q_2 \pmod n$ est non trivial ; il révèle la même décomposition que q_1.
- Lorsque q_1 et q_2 sont non triviaux et $\Delta_1 = \pm\Delta_2$, le nombre composé $q_1\times q_2$ (mod n) est trivial ; il ne révèle pas de décomposition.
- Lorsque q_1 et q_2 sont non triviaux et $\Delta_1 \neq \pm\Delta_2$, le nombre composé $q_1\times q_2$ (mod n) est non trivial ; il révèle une troisième décomposition.

Trois nombres $\{q_1, q_2, q_3\}$ donnent quatre nombres composés $\{q_1\times q_2, q_1\times q_3, q_2\times q_3, q_1\times q_2\times q_3 \pmod n\}$, soit un total de sept nombres ; m nombres donnent ainsi 2^m-m-1 nombres composés, soit un total de 2^m-1 nombres.

Considérons un jeu de clés GQ2 généralisées comportant i nombres de base g_1 à g_i et i nombres privés Q_1 à Q_i donnant i nombres q_1 à q_i et donc i nombres Δ_1 à Δ_i qui sont des racines de l'unité. Cherchons à prendre en compte un autre nombre de base g_{i+1} par un nombre privé Q_{i+1} donnant un nombre q_{i+1} et donc une racine Δ_{i+1}.

• Le total des $2^{i+1}-1$ nombres comporte autant de nombres non triviaux dans chacun des deux cas suivants.

- La racine Δ_{i+1} est triviale et au moins une racine Δ_1 à Δ_i est non triviale.
- La racine Δ_{i+1} est non triviale et figure parmi les $2\times i$ racines $\pm\Delta_1$ à $\pm\Delta_i$.

• Dans le cas où la racine Δ_{i+1} est non triviale et ne figure pas parmi les $2\times i$ racines $\pm\Delta_1$ à $\pm\Delta_i$, chaque nombre composé où figure q_{i+1} est non trivial.

Par conséquent, lorsque parmi m nombres q_1 à q_m, au moins un est non trivial, plus de la moitié du total des 2^m-1 nombres sont non triviaux.

Par définition, nous disons que $l < f$ nombres non triviaux $\{q_1, q_2, \ldots q_l\}$ sont <u>indépendants</u> par rapport au module n lorsque chacun des 2^l-l-1

nombres composés correspondants est non trivial, c'est-à-dire que, au total, les 2^l-1 nombres sont tous non triviaux. Chacun de ces 2^l-1 nombres révèle alors une décomposition différente du module n.

- Lorsque les f facteurs premiers sont distincts, il y a $2^{f-1}-1$ décompositions du module n. Alors, si $f-1$ nombres q sont indépendants, il y a une correspondance biunivoque entre les $2^{f-1}-1$ décompositions et un total de $2^{f-1}-1$ nombres comprenant les $f-1$ nombres indépendants et les $2^{f-1}-f$ nombres composés correspondants.

Restes chinois — Soient deux nombres a et b premiers entre eux tels que $0 < a < b$, et deux nombres X_a de 0 à $a-1$ et X_b de 0 à $b-1$; il s'agit de déterminer le nombre unique X de 0 à $a \times b-1$ tel que $X_a \equiv X \pmod a$ et $X_b \equiv X \pmod b$. Le nombre $\alpha \equiv \{b \pmod a\}^{-1} \pmod a$ est le paramètre des restes chinois. Voici l'opération élémentaire des restes chinois.

$x \equiv X_b \pmod a$

$y = X_a - x$; si y est négatif, remplacer y par $y+a$

$z \equiv \alpha \times y \pmod a$

$X = z \times b + X_b$

En résumé, nous écrivons : $X =$ Restes Chinois (X_a, X_b).

Lorsque f facteurs premiers sont rangés dans l'ordre croissant, du plus petit p_1 au plus grand p_f, les paramètres des restes chinois peuvent être les suivants (il y en a un de moins que de facteurs premiers, c'est-à-dire $f-1$).

- Le premier paramètre est $\alpha \equiv (p_2 \pmod{p_1})^{-1} \pmod{p_1}$.
- Le second paramètre est $\beta \equiv (p_1 \times p_2 \pmod{p_3})^{-1} \pmod{p_3}$.
- Le i-ième paramètre est $\lambda \equiv (p_1 \times \ldots p_{i-1} \pmod{p_i})^{-1} \pmod{p_i}$.
- Et ainsi de suite.

En $f-1$ opérations élémentaires, on établit un nombre X de 0 à $n-1$ à partir de tout jeu de f composantes de X_1 à X_f avec X_j de 0 à p_j-1 :

- un premier résultat $(\text{mod } p_1 \times p_2)$ avec le premier paramètre,
- puis, un second résultat $(\text{mod } p_1 \times p_2 \times p_3)$ avec le second paramètre,

- jusqu'au résultat final (mod $n = p_1 \times p_2 \times \ldots p_f$) avec le dernier paramètre.

En résumé, étant donnés les facteurs premiers p_1 à p_f, chaque élément de l'anneau des entiers modulo n a deux représentations équivalentes :

- f nombres X_1 à X_f, une composante par facteur premier : $X_j \equiv X \pmod{p_j}$,
- un nombre X de 0 à $n-1$, $X =$ Restes Chinois $(X_1, X_2, \ldots X_f)$.

Rang des nombres dans CG(p) — Soit un nombre premier impair p et un nombre a plus petit que p, c'est-à-dire $0 < a < p$. Par définition, le rang de a par rapport à p est la période de la suite $\{X\}$ définie par $\{x_1 = a$; puis, pour $i \geq 1$, $x_{i+1} \equiv a \times x_i \pmod{p}\}$. Grâce au théorème de Fermat, nous obtenons : $x_{i+p} \equiv a^p \times x_i \equiv a \times x_i \equiv x_{i+1} \pmod{p}$. Par conséquent, le rang d'un nombre a par rapport à un nombre premier p est $p-1$ ou un diviseur de $p-1$.

Par exemple, lorsque $(p-1)/2$ est un nombre premier impair p', le corps de Galois CG(p) comporte un nombre de rang 1 : c'est 1, un nombre de rang 2 : c'est -1, $p'-1$ nombres de rang p' et $p'-1$ nombres de rang $2 \times p' = p-1$.

Dans CG(p), tout nombre de rang $p-1$ est un « générateur ». La dénomination est due au fait que les puissances successives d'un générateur dans CG(p), c'est-à-dire les termes de la suite $\{X\}$ pour les indices de 1 à $p-1$, forment une permutation de tous les éléments non nuls de CG(p).

Soit un générateur y de CG(p). Evaluons le rang du nombre $y^i \pmod{p}$ en fonction de i et de $p-1$. Lorsque i est premier avec $p-1$, c'est $p-1$. Lorsque i divise $p-1$, c'est $(p-1)/i$. Dans tous les cas, c'est $(p-1)/\text{pgcd}(p-1, i)$.

Par définition, la fonction d'Euler $\varphi(n)$ est le nombre de nombres plus petits que n et premiers avec n. Dans CG(p), il y a $\varphi(p-1)$ générateurs.

A titre d'illustration, le rang fait bien comprendre les bases du RSA. Le module n est le produit de f facteurs premiers p_1 à p_f avec $f \geq 2$. Pour chaque facteur premier p_j de p_1 à p_f, l'exposant public e doit être premier avec p_j-1. Alors, la clé $\langle e, p_j \rangle$ respecte le rang des éléments de CG(p_j) : elle permute les éléments de CG(p_j) ; il existe un nombre d_j, généralement le plus petit

possible, tel que p_j-1 divise $e\times d_j-1$. La clé $\langle d_j, p_j\rangle$ inverse la permutation des éléments de CG(p_j). Ces f permutations, une dans chaque corps CG(p_1) à CG(p_f), se traduisent dans l'anneau des entiers modulo n par la permutation RSA résumée par la clé publique $\langle e, n\rangle$. Il existe un nombre d, généralement le plus petit possible, tel que ppcm($p_1-1, p_2-1, \ldots p_f-1$) divise $d\times e-1$. Pour chaque facteur premier p_j de p_1 à p_f, on a $d_j \equiv d \pmod{p_j-1}$. La permutation RSA résumée par la clé publique $\langle e, n\rangle$ est inversée par la clé privée $\langle d, n\rangle$.

Carrés dans CG(p) — Définissons un nombre t tel que $p-1$ est divisible par 2^t, mais pas par 2^{t+1}. Chaque grand nombre premier figure dans une et une seule catégorie : $t=1$, $t=2$, $t=3$, $t=4$, et ainsi de suite. Si l'on considère un assez grand nombre de nombres premiers successifs, environ un sur deux figure dans la première catégorie où p est congru à 3 (mod 4), un sur quatre dans la deuxième où p est congru à 5 (mod 8), un sur huit dans la troisième où p est congru à 9 (mod 16), un sur seize dans la quatrième où p est congru à 17 (mod 32), et ainsi de suite ; en moyenne, un sur 2^t figure dans la t-ième catégorie où p est congru à $2^t+1 \pmod{2^{t+1}}$.

Parce que les nombres x et $p-x$ ont le même carré dans CG(p), la clé $\langle 2, p\rangle$ ne permute pas CG(p). La fonction « élever au carré » dans CG(p) peut se représenter par un graphe orienté où chaque élément non nul du corps trouve sa place. Analysons la structure du graphe en branches et en cycles selon la parité du rang de chaque élément.

- <u>L'élément nul est fixe</u>. C'est 0. Le rang n'est pas défini pour l'élément nul auquel aucun autre élément ne se rattache ; l'élément nul est isolé.
- <u>L'élément unité est fixe</u>. C'est 1, le seul élément de rang 1. Toutes les racines de l'unité dans CG(p) se trouvent dans la branche se rattachant à 1. Soit y un résidu non quadratique de CG(p), n'importe lequel ; la clé $\langle (p-1)/2^t, p\rangle$ transforme y en une racine 2^{t-1}-ième primitive de -1 notée par b ; en effet, on a $y^{(p-1)/2} \equiv -1 \pmod{p}$. Par conséquent, dans CG(p), les puissances de b pour les exposants de 1 à 2^{t-1} sont les 2^{t-1} racines de

l'unité autres que 1 : elles composent la branche se rattachant à 1.

- Le carré de tout élément de rang pair est un autre élément dont le rang est divisé par deux. Par conséquent, chaque élément de rang pair se place dans une branche ; chaque branche comporte un nombre de rang divisible par deux mais pas par quatre, puis, si $t \geq 2$, deux nombres de rang divisible par quatre mais pas par huit, puis, si $t \geq 3$, quatre nombres de rang divisible par huit mais pas par seize, puis, si $t \geq 4$, huit nombres de rang divisible par seize mais pas par 32, et ainsi de suite. Toutes les branches sont semblables à la branche rattachée à 1 ; les 2^{t-1} feuilles de chaque branche sont des résidus non quadratiques ; chaque branche comporte 2^t-1 éléments et se rattache à un élément de rang impair ; il y a $(p-1)/2^t$ branches qui ont toutes la même longueur t.
- Le carré de tout élément de rang impair autre que l'élément unité est un autre élément ayant le même rang. La clé $\langle 2, p\rangle$ permute l'ensemble des $(p-1)/2^t$ éléments de rang impair. La permutation se décompose en cycles de permutation. Le nombre de cycles dépend de la factorisation de $(p-1)/2^t$. Pour chaque diviseur p' de $(p-1)/2^t$, il y a un cycle comportant les $\varphi(p')$ éléments de rang p'. Rappelons que par définition, la fonction d'Euler $\varphi(p')$ est le nombre de nombres plus petits que p' et premiers avec p'. Par exemple lorsque $p' = (p-1)/2^t$ est premier, les $p'-1$ nombres de rang p' forment un grand cycle de permutation.

Les figures 1A à 1D illustrent chacune un fragment de graphe pour p congru respectivement à 3 (mod 4), 5 (mod 8), 9 (mod 16) et 17 (mod 32).

- Les feuilles sur les branches sont représentées par des ronds blancs ; ce sont des résidus non quadratiques.
- Les nœuds dans les branches sont représentés par des ronds gris ; ce sont des éléments quadratiques de rang pair.
- Les nœuds dans les cycles sont représentés par des ronds noirs ; ce sont des éléments quadratiques de rang impair.

Racines carrées dans CG(p) — Sachant que a est un résidu quadratique de CG(p), voyons comment calculer une solution à l'équation $x^2 \equiv a \pmod p$, c'est-à-dire « prendre une racine carrée » dans CG(p). Il y a bien sûr plusieurs façons d'obtenir le même résultat : on pourra consulter les pages 31 à 36 du livre de Henri Cohen, *a Course in Computational Algebraic Number Theory*, publié en 1993 par Springer à Berlin comme volume 138 de la série *Graduate Texts in Mathematics* (GTM 138).

Le nombre $s = (p-1+2^t)/2^{t+1}$ donne une clé $\langle s, p\rangle$ qui vaut :

$\langle (p+1)/4, p\rangle$ lorsque p est congru à 3 (mod 4),
$\langle (p+3)/8, p\rangle$ lorsque p est congru à 5 (mod 8),
$\langle (p+7)/16, p\rangle$ lorsque p est congru à 9 (mod 16),
$\langle (p+15)/32, p\rangle$ lorsque p est congru à 17 (mod 32),
et ainsi de suite.

- La clé $\langle s, p\rangle$ transforme tout élément d'un cycle en l'élément précédent dans le cycle. Lorsque a est de rang impair, c'est la solution de rang impair ; nous la nommons w. En effet, dans CG(p), w^2/a vaut a élevé à la puissance $(2\times(p-1+2^t)/2^{t+1})-1 = (p-1)/2^t$. L'autre solution est de rang pair ; c'est $p-w$.

- D'une manière générale, la clé $\langle s, p\rangle$ transforme tout résidu quadratique a en une première approximation de solution que nous nommons r. Puisque a est un résidu quadratique, la clé $\langle 2^{t-1}, p\rangle$ transforme certainement r^2/a en 1. Pour se rapprocher d'une racine carrée de a, élevons r^2/a à la puissance 2^{t-2} (mod p) pour obtenir +1 ou −1. La nouvelle approximation reste r si le résultat est +1 ou bien devient $b\times r \pmod p$ si le résultat est −1, sachant que b désigne n'importe quelle racine 2^t-ième primitive de 1 dans le corps CG(p). Par conséquent, la clé $\langle 2^{t-2}, p\rangle$ transforme la nouvelle approximation en 1. On peut encore se rapprocher en utilisant la clé $\langle 2^{t-3}, p\rangle$ et en multipliant par $b^2 \pmod p$ s'il le faut, et ainsi de suite.

L'algorithme suivant résout l'équation. Il utilise les nombres a, b, p, r et t définis ci-dessus et deux variables : c représente les corrections successives

et w les approximations successives. Au début de l'algorithme, $c = b$ et $w = r$. A l'issue du calcul, les deux solutions sont w et $p-w$.

Pour i allant de $t-2$ à 1, répéter la séquence suivante :

- Appliquer la clé $\langle 2^i, p\rangle$ au nombre $w^2/a \pmod p$ pour obtenir +1 ou −1.
- Lorsque l'on obtient −1, remplacer w par $w \times c \pmod p$.
- Remplacer c par $c^2 \pmod p$.

Applicabilité des principes — Par définition, nous disons qu'un paramètre k, un nombre de base g et un facteur premier p sont *compatibles* lorsque l'équation $x^v \equiv g^2 \pmod p$ où l'exposant v vaut 2^k a des solutions en x dans le corps CG(p). Les nombres k et g sont petits et plus grands que 1. Le nombre p est un grand nombre premier.

- Lorsque $t = 1$, c'est-à-dire $p \equiv 3 \pmod 4$, l'équation a deux solutions.

- Lorsque $t = 2$, c'est-à-dire $p \equiv 5 \pmod 8$, selon le symbole de Legendre de g par rapport à p, l'équation a quatre solutions si $(g|p) = +1$; elle n'a pas de solution si $(g|p) = -1$.

- Lorsque $t > 2$, c'est-à-dire $p \equiv 1 \pmod 8$, soit u le nombre tel que 2^u divise le rang du nombre public $G = g^2$ par rapport à p, mais que 2^{u+1} ne le divise pas ; par conséquent, u est égal à l'un des nombres de 0 à $t-1$. L'équation n'a aucune solution si $u > 0$ et $k+u > t$; elle a 2^k solutions si $k+u \leq t$; elle a 2^t solutions si $u = 0$ et $k > t$.

Il y a donc deux types de compatibilité selon que G est dans un cycle ou bien en position appropriée dans une branche.

- Lorsque G est dans un cycle, c'est-à-dire $u = 0$ quelle que soit la valeur de k, il y a une solution de rang impair dans le cycle et des solutions de rang pair disséminées dans $\alpha = \min(k, t)$ branches consécutives rattachées au cycle, soit 2^α solutions en tout. La figure 2A illustre ce cas avec $k \geq t = 3$, c'est-à-dire un facteur premier congru à 9 (mod 16), ce qui impose $u = 0$.
- Lorsque G est en position appropriée dans une branche, c'est-à-dire

$u > 0$ et $u+k \leq t$, il y a 2^k solutions, toutes de rang pair et dans la branche. La figure 2B illustre ce cas.

Etant donné un paramètre k, il y a donc deux types de facteurs premiers selon que la valeur de t est inférieure à k ou bien supérieure ou égale à k.

- Pour tout facteur premier p_j tel que $t < k$, chaque G_i doit être dans un cycle et il n'y a pas de solution dans la branche rattachée à G_i. Définissons un nombre $\Delta_{i,j}$ qui vaut +1 ou −1 selon que g_i ou $-g_i$ est dans le cycle. Il n'y a pas de choix pour aucun des m nombres $\Delta_{1,j}$ à $\Delta_{m,j}$. La figure 3A illustre un cas $t < k$: G_i est dans un cycle avec un facteur premier p_j congru à 9 (mod 16), c'est-à-dire, $u = 0$, $t = 3$ avec $k > 3$.
- Pour tout facteur premier p_j tel que $t \geq k$, chaque G_i doit être tel que $u+k \leq t$, c'est-à-dire, ou bien dans un cycle avec $u = 0$ ou bien en position appropriée dans une branche avec $1 \leq u \leq t-k$. Définissons un nombre $\Delta_{i,j}$ qui vaut +1 ou −1 selon que $Q_{i,j}$ se trouve dans la partie de graphe rattachée à g_i ou à $-g_i$. Il y a le choix pour chacun des m nombres $\Delta_{1,j}$ à $\Delta_{m,j}$; chaque nombre $\Delta_{i,j}$ peut être individuellement basculé d'une valeur à l'autre. La figure 3B illustre un cas $t \geq k$: G_i est dans une branche avec un facteur premier p_j congru à 17 (mod 32), c'est-à-dire, $u = 1$, $t = 4$ avec $k = 3$.

Chaque jeu de f composantes $\{\Delta_{i,1} \ldots \Delta_{i,f}\}$ est une racine carrée de l'unité dans $CG(p_j)$. Cette racine est triviale ou pas selon que les f composantes sont égales ou pas ; nous disons alors que le jeu de f composantes est constant ou variable, ce qui traduit le fait que le nombre q_i est trivial ou pas. Par conséquent, lorsqu'un nombre q_i est non trivial, le jeu de f composantes $\{\Delta_{i,1} \ldots \Delta_{i,f}\}$ résume une décomposition du module. Il est donc possible de tester les principes avant de calculer les composantes privées $Q_{i,j}$.

- Lorsqu'un nombre public G_i est dans un cycle pour un facteur premier p_j, le nombre $\Delta_{i,j}$ vaut +1 ou −1 selon que g_i ou $-g_i$ est dans le cycle. Lorsque $p_j \equiv 3 \pmod 4$, c'est le symbole de Legendre : $\Delta_{i,j} = (g_i | p_j)$.

- Lorsqu'un nombre public G_i est en position appropriée dans une branche pour un facteur premier p_j, on peut déterminer la valeur à donner à $\Delta_{i,j}$ avant de calculer la composante privée $Q_{i,j}$.

Production de jeux de clés — Etant donné un paramètre k, il y a deux stratégies.

- Ou bien le générateur demande f facteurs premiers afin de déterminer m nombres de base. Les premiers nombres premiers : 2, 3, 5, 7, ... sont examinés pour évaluer leur compatibilité avec chacun des f grands facteurs premiers p_1 à p_f. Bien que $g = 2$ ne soit pas compatible avec $p \equiv 5 \pmod 8$, 2 peut entrer dans la composition d'un nombre de base. En effet, lorsque deux nombres sont en position similaire dans une branche, leur produit est plus près du cycle, tout comme un carré rapproche du cycle. On peut ainsi obtenir un nombre de base en composant des nombres qui individuellement ne conviennent pas.
- Ou bien le générateur demande m nombres de base et des caractéristiques du module telle qu'une taille en bits (par exemple, 512, 768, 1024, 1536, 2048) et un nombre de bits successifs à 1 en poids forts (par exemple, 1, 8, 16, 24, 32) afin de déterminer $f \geq 2$ facteurs premiers. Notés par g_1 g_2 ... g_m, les nombres de base figurent généralement parmi les premiers nombres premiers : 2, 3, 5, 7, 11, ... ou bien ce sont des combinaisons des premiers nombres premiers. Faute d'indication contraire, ce sont les m premiers nombres premiers : $g_1 = 2$, $g_2 = 3$, $g_3 = 5$, $g_4 = 7$, ... Notons que $p \equiv 5 \pmod 8$ n'est pas compatible avec $g = 2$. Le module n sera le produit de f facteurs premiers de tailles voisines, à savoir la taille assignée au module divisée par f.

Premier principe — Le paramètre k, chaque facteur premier p allant de p_1 à p_f et chaque nombre de base g allant de g_1 à g_m doivent être compatibles. Définissons un nombre h tel que 2^h divise le rang de g par rapport à p, alors que 2^{h+1} ne le divise pas. Pour calculer le nombre h, la procédure suivante

utilise le symbole de Legendre $(g|p)$ et un nombre b, racine 2^t-ième primitive de l'unité dans CG(p).

- Si $(g|p) = +1$ avec $t = 1$, retourner « $h = 0$ ».
- Si $(g|p) = +1$ avec $t > 1$, appliquer la clé $\langle(p-1+2^t)/2^{t+1}, p\rangle$ à G pour obtenir un résultat appelé w.
  - Si $w = +g$, retourner « $h = 0$ ».
  - Si $w = p-g$, retourner « $h = 1$ ».
  - Sinon, mettre c à b et pour i allant de $t-1$ à 2,
    - appliquer la clé $\langle 2^i, p\rangle$ à $w/g \pmod p$ pour obtenir ± 1,
    - si -1, mettre h à i et remplacer w par $w\times c \pmod p$,
    - remplacer c par $c^2 \pmod p$.
  - Retourner « valeur de h de 2 à $t-1$ ».
- Si $(g|p) = -1$, retourner « $h = t$ ».

Rappelons que k, g et p sont incompatibles lorsque $u > 0$ avec $k+u > t$; ils sont compatibles lorsque $h = 0$ ou 1, quelle que soit la valeur de k, et également lorsque $h > 1$ avec $k+h \leq t+1$.

Second principe — Les trois procédures suivantes correspondent à différentes implémentations du second principe. Dans certaines implémentations, le second principe peut être renforcé au point d'exiger que chaque nombre q_1 à q_m soit non trivial. Le rôle des nombres de base est alors équilibré ; le fait d'équilibrer ou pas le second principe a un effet sur certains aspects de démonstration de la sécurité du schéma. Enfin, lorsqu'il y a $f > 2$ facteurs premiers distincts, parmi les m nombres $\{q_1 \ldots q_m\}$, on peut exiger qu'il y ait au moins un sous ensemble de $f-1$ nombres indépendants.

Les trois procédures utilisent $m\times f$ nombres $\delta_{i,j}$ définis comme suit.

- Lorsque p_j est tel que $t < k$, pour i allant de 1 à m, $\delta_{i,j} = \Delta_{i,j}$, c'est-à-dire $+1$ si $h_{i,j} = 0$ et -1 si $h_{i,j} = 1$.
- Lorsque p_j est tel que $t \geq k$, pour i allant de 1 à m, $\delta_{i,j} = 0$, ce qui indique que $\Delta_{1,j}$ à $\Delta_{m,j}$ peuvent être choisis en fonction du deuxième principe.

Une première procédure vérifie qu'au moins un jeu $\{\delta_{i.1} \ldots \delta_{i,f}\}$ est variable ou nul, c'est-à-dire qu'au moins un nombre q_1 à q_m est non trivial ou peut être choisi non trivial.

- Pour i allant de 1 à m et j allant de 1 à f,
  - si $\delta_{i,j} = 0$ ou $\neq \delta_{i.1}$, retourner « succès ».
- Retourner « échec ».

Une deuxième procédure vérifie que chaque jeu $\{\delta_{i.1} \ldots \delta_{i,f}\}$ est variable ou nul, c'est-à-dire que chaque nombre q_1 à q_m est non trivial ou peut être choisi non trivial.

- Pour i allant de 1 à m,
  - pour j allant de 1 à f,
    - si $\delta_{i,j} = 0$ ou $\neq \delta_{i.1}$, passer à la valeur suivante de i.
  - Retourner « échec ».
- Retourner « succès ».

Une troisième procédure vérifie que pour chaque paire de facteurs premiers p_{j1} et p_{j2} avec $1 \leq j_1 < j_2 \leq f$, il y a au moins un jeu $\{\delta_{i.1} \ldots \delta_{i,f}\}$ où $\delta_{i,j1}$ est nul ou différent de $\delta_{i,j2}$. Elle échoue manifestement lorsque m est plus petit que $f-1$. Lorsqu'elle réussit, parmi les m nombres q_1 à q_m, il y a au moins un ensemble de $f-1$ nombres indépendants par rapport aux f facteurs premiers.

- Pour j_1 allant de 1 à $f-1$ et pour j_2 allant de j_1+1 à f,
  - pour i allant de 1 à m,
    - si $\delta_{i,j1} = 0$ ou $\neq \delta_{i,j2}$, passer aux valeurs suivantes de j_1 et j_2.
  - Retourner « échec ».
- Retourner « succès ».

Lorsqu'une procédure échoue, le générateur de jeux de clés GQ2 suit sa stratégie parmi les deux stratégies possibles :

- changer l'un des m nombres de base en gardant les f facteurs premiers,
- changer l'un des f facteurs premiers en gardant les m nombres de base.

Troisième principe — La procédure suivante détermine si le jeu de clés

GQ2 généralisées en cours de production ou déjà produit est

- un jeu de clés GQ2 élémentaires, c'est-à-dire que les $2\times m$ nombres $\pm g_1$ à $\pm g_m$ sont tous des résidus non quadratiques,
- ou bien, un jeu de clés GQ2 complémentaires, c'est-à-dire que parmi les $2\times m$ nombres $\pm g_1$ à $\pm g_m$, il y a au moins un résidu quadratique.

La procédure utilise les deux symboles de Legendre $(g_i \mid p_j)$ et $(-g_i \mid p_j)$ pour i allant de 1 à m et pour j allant de 1 à f.

- Pour i allant de 1 à m,
  - pour j allant de 1 à f,
    - si $(g_i \mid p_j) = -1$, passer à la valeur suivante de i.
  - Retourner « jeu de clés GQ2 complémentaires ».
  - pour j allant de 1 à f,
    - si $(-g_i \mid p_j) = -1$, passer à la valeur suivante de i.
  - Retourner « jeu de clés GQ2 complémentaires ».
- Retourner « jeu de clés GQ2 élémentaires ».

Composantes privées — Pour une équation de type direct : $x^v \equiv g_i^2 \pmod{p_j}$, les calculs suivants établissent toutes les valeurs possibles de la composante privée $Q_{i,j}$. Les deux cas les plus simples et les plus courants, c'est-à-dire $t = 1$ et $t = 2$, sont suivis par le cas plus complexe, c'est-à-dire $t > 2$.

Pour $t = 1$, c'est-à-dire $p_j \equiv 3 \pmod 4$), la clé $\langle (p_j+1)/4, p_j \rangle$ donne la racine carrée quadratique de n'importe quel résidu quadratique dans $CG(p_j)$. On en déduit un nombre $s_j \equiv ((p_j+1)/4)^k \pmod{(p_j-1)/2}$, ce qui donne une clé $\langle s_j, p_j \rangle$ transformant G_i en $w \equiv G_i^{sj} \pmod{p_j}$. $Q_{i,j}$ est égal à w ou bien à p_j-w.

Pour $t = 2$, c'est-à-dire $p_j \equiv 5 \pmod 8$), la clé $\langle (p_j+3)/8, p_j \rangle$ donne la racine carrée de rang impair de n'importe quel élément de rang impair dans $CG(p_j)$. On en déduit un nombre $s_j \equiv ((p_j+3)/8)^k \pmod{(p_j-1)/4}$, ce qui donne une clé $\langle s_j, p_j \rangle$ transformant G_i en $w \equiv G_i^{sj} \pmod{p_j}$. Remarquons que $z \equiv 2^{(pj-1)/4} \pmod{p_j}$ est une racine carrée de -1 parce que 2 est un résidu non quadratique dans $CG(p_j)$. $Q_{i,j}$ est égal à w ou bien à p_j-w ou bien encore à

$w' \equiv w \times z \pmod{p_j}$ ou bien à $p_j - w'$.

Pour $p_j \equiv 2^t+1 \pmod{2^{t+1}}$ avec $t > 2$, la clé $\langle (p_j-1+2^t)/2^{t+1}, p_j \rangle$ donne la racine carrée de rang impair de n'importe quel élément de rang impair. Le test de compatibilité entre k, g et p a donné la valeur de h, puis celle de u.

- Lorsque G_i est dans un cycle ($u = 0$, quelle que soit la valeur de k), on établit un nombre $s_j \equiv ((p_j-1+2^t)/2^{t+1})^k \pmod{(p_j-1)/2^t}$. La clé $\langle s_j, p_j \rangle$ transforme G_i en la solution de rang impair $w \equiv G_i^{sj} \pmod{p_j}$. Il y des solutions de rang pair réparties dans $\min(k, t)$ branches consécutives rattachées au cycle, disons dans α branches. $Q_{i,j}$ est égal au produit de w par n'importe laquelle des racines 2^α-ièmes de l'unité dans $CG(p_j)$.
- Lorsque G_i est en position appropriée dans une branche ($u > 0$, $u+k \leq t$), toutes les solutions sont dans la même branche que G_i, branche rattachée à un cycle par la puissance 2^u-ième du nombre G_i. On établit un nombre $s_j \equiv ((p_j-1+2^t)/2^{t+1})^{k+u} \pmod{(p_j-1)/2^t}$. La clé $\langle s_j, p_j \rangle$ transforme la puissance 2^u-ième de G_i en un nombre de rang impair w. L'ensemble des produits de w par les racines 2^{k+u}-ièmes <u>primitives</u> de l'unité dans $CG(p_j)$ comprend les 2^k valeurs de $Q_{i,j}$.

Lorsque p_j est tel que $t \geq k$, le nombre b_j étant une racine 2^t-ième primitive de l'unité dans $CG(p_j)$, la puissance 2^{t-u}-ième de b_j dans $CG(p_j)$ existe ; c'est une racine 2^k-ième primitive de l'unité. Multiplier $Q_{i,j}$ par une racine 2^k-ième primitive de l'unité permet de basculer la valeur du nombre $\Delta_{i,j}$.

Pour une équation de type inverse : $1 \equiv x^v \times g_i^2 \pmod{p_j}$, il suffit de remplacer le nombre s_j par $((p_j-1)/2^t)-s_j$ dans la clé $\langle s_j, p_j \rangle$, ce qui revient à inverser la valeur de $Q_{i,j}$ dans $CG(p_j)$.

Exemple de jeu de clés à deux facteurs premiers congrus à 5 (mod 8)

p_1 = E6C83BF428689AF8C35E07EDD06F9B39A659829A58B79CD894C435C95F32BF25

p_2 = 11BF8A68A0817BFCC00F15731C8B70CEF9204A34133A0DEF862829B2EEA74873D

$n = p_1 \times p_2$ = FFFF8263434F173D0F2E76B32D904F56F4A5A6A50008C43
D32B650E9AB9AAD2EB713CD4F9A97C4DBDA3828A3954F296458D5
F42C0126F5BD6B05478BE0A80ED1

Voici les symboles de Legendre des tout premiers nombres premiers.

$(2 \mid p_1) = -1$; $(3 \mid p_1) = -1$; $(5 \mid p_1) = +1$; $(7 \mid p_1) = -1$;
$(11 \mid p_1) = +1$; $(13 \mid p_1) = -1$; $(17 \mid p_1) = +1$;

Dans CG(p_1), le rang est impair pour -5, -11 et 17.

$(2 \mid p_2) = -1$; $(3 \mid p_2) = +1$; $(5 \mid p_2) = +1$; $(7 \mid p_2) = +1$;
$(11 \mid p_2) = +1$; $(13 \mid p_2) = -1$; $(17 \mid p_2) = -1$;

Dans CG(p_2), le rang est impair pour 3, -5, 7 et 11.

La fonction de Carmichael est $\lambda(n) = \text{ppcm}((p_1-1)/4, (p_2-1)/4)$.

$\lambda(n)$ = 33331A13DA4304A5CFD617BD6F83431164212154334F40C3D5
7A9C8558555D5BDAA2EF6AED17B9E3794F51A65A1B37239B18FA9
B0F618627D8C7E1D8499C1B

Avec $k = 9$, on utilise le nombre $\sigma \equiv \lambda(n) - ((1+\lambda(n))/2)^9 \pmod{\lambda(n)}$ comme exposant privé, de façon à utiliser des équations génériques de type inverse.

σ = 01E66577BC997CAC273671E187A35EFD25373ABC9FE6770E7446
C0CCEF2C72AF6E89D0BE277CC6165F1007187AC58028BD2416D4CC
1121E7A7A8B6AE186BB4B0

Les nombres 2, 3, 7, 13 et 17 ne conviennent pas comme nombre de base.

La clé $\langle \sigma, n \rangle$ transforme $g_1 = 5$ en un nombre privé Q_1 qui ne révèle pas de décomposition. En effet, dans les deux corps, -5 est sur un cycle.

Q_1 = 818C23AF3DE333FAECE88A71C4591A70553F91D6C0DD5538EC
0F2AAF909B5BDAD491FD8BF13F18E3DA3774CCE19D0097BC4BD4
7C5D6E0E7EBF6D89FE3DC5176C

La clé $\langle \sigma, n \rangle$ transforme $g_2 = 11$ en un nombre privé Q_2 qui révèle une décomposition. En effet, 11 n'est pas en même position dans les deux corps.

Q_2 = 25F9AFDF177993BE8652CE6E2C728AF31B6D66154D3935AC535
196B07C19080DC962E4E86ACF40D01FDC454F2565454F290050DA05

2089EEC96A1B7DEB92CCA7

La clé $\langle\sigma, n\rangle$ transforme $g_3 = 21 = 3\times7$ en un nombre privé Q_3 qui révèle une décomposition.

Q_3 = 78A8A2F30FEB4A5233BC05541AF7B684C2406415EA1DD67D18 A0459A1254121E95D5CAD8A1FE3ECFE0685C96CC7EE86167D99532 B3A96B6BF9D93CAF8D4F6AF0

La clé $\langle\sigma, n\rangle$ transforme $g_4 = 26 = 2\times13$ en un nombre privé Q_4 qui révèle une décomposition.

Q_4 = 6F1748A6280A200C38824CA34C939F97DD2941DAD300030E481 B738C62BF8C673731514D1978AF5655FE493D659514A6CE897AB76C 01E50B5488C5DAD12332E5

La clé privée peut encore se représenter par les deux facteurs premiers, le paramètre des restes chinois et huit composantes privées.

$\alpha \equiv (p_2 \pmod{p_1})^{-1} \pmod{p_1}$ = ADE4E77B703F5FDEAC5B9AAE825D649 E06692D15FBF0DF737B115DC4D012FD1D

$Q_{1,1} \equiv Q_1 \pmod{p_1}$ = 7751A9EE18A8F5CE44AD73D613A4F465E06C6F9 AF4D229949C74DD6C18D76FAF

$Q_{1,2} \equiv Q_1 \pmod{p_2}$ = A9EB5FA1B2A981AA64CF88C382923DB64376F5F D48152C08EEB6114F31B7665F

$Q_{2,1} \equiv Q_2 \pmod{p_1}$ = D5A7D33C5FB75A033F2F0E8B20274B957FA3400 4ABB2C2AC1CA3F5320C5A9049

$Q_{2,2} \equiv Q_2 \pmod{p_2}$ = 76C9F5EFD066C73A2B5CE9758DB512DFC011F5B 5AF7DA8D39A961CC876F2DD8F

$Q_{3,1} \equiv Q_3 \pmod{p_1}$ = 2FEC0DC2DCA5BA7290B27BC8CC85C938A514B 8F5CFD55820A174FB5E6DF7B883

$Q_{3,2} \equiv Q_3 \pmod{p_2}$ = 010D488E6B0A38A1CC406CEE0D55DE59013389D 8549DE493413F34604A160C1369

$Q_{4,1} \equiv Q_4 \pmod{p_1}$ = A2B32026B6F82B6959566FADD9517DB8ED85246 52145EE159DF3DC0C61FE3617

$Q_{4,2} \equiv Q_4 \pmod{p_2}$ = 011A3BB9B607F0BD71BBE25F52B305C224899E5 F1F8CDC2FE0D8F9FF62B3C9860F

Polymorphisme de la clé privée GQ2 — Les diverses représentations possibles de la clé privée GQ2 s'avèrent équivalentes : elles se ramènent toutes à la connaissance de la factorisation du module n qui est la véritable clé privée GQ2. La représentation de la clé privée GQ2 a un effet sur le déroulement des calculs au sein de l'entité qui prouve, pas au sein de l'entité qui contrôle. Voici les trois principales représentations possibles de la clé privée GQ2. 1) La représentation classique des clés privées GQ consiste à stocker m nombres privés Q_i et la clé publique de vérification $\langle v, n \rangle$; pour les schémas GQ2, cette représentation est concurrencée par les deux suivantes. 2) La représentation optimale en termes de charges de travail consiste à stocker le paramètre k, les f facteurs premiers p_j, $m \times f$ composantes privées $Q_{i,j}$ et $f-1$ paramètres des restes chinois. 3) La représentation optimale en termes de taille de clé privée consiste à stocker le paramètre k, les m nombres de base g_i et les f facteurs premiers p_j, puis, à commencer chaque utilisation en établissant ou bien m nombres privés Q_i et le module n pour se ramener à la première représentation, ou bien $m \times f$ composantes privées $Q_{i,j}$ et $f-1$ paramètres des restes chinois pour se ramener à la seconde.

Parce que la sécurité du mécanisme d'authentification dynamique ou de signature numérique équivaut à la connaissance d'une décomposition du module, les schémas GQ2 ne permettent pas de distinguer simplement deux entités utilisant le même module. Généralement, chaque entité qui prouve dispose de son propre module GQ2. Toutefois, on peut spécifier des modules GQ2 à quatre facteurs premiers dont deux sont connus d'une entité et les deux autres d'une autre.

Authentification dynamique — Le mécanisme d'authentification dynamique est destiné à prouver à une entité appelée **contrôleur** l'authenticité

d'une autre entité appelée **démonstrateur** ainsi que l'authenticité d'un éventuel message associé M, de sorte que le contrôleur s'assure qu'il s'agit bien du démonstrateur et éventuellement que lui et le démonstrateur parlent bien du même message M. Le message associé M est optionnel, ce qui signifie qu'il peut être vide.

Le mécanisme d'authentification dynamique est une séquence de quatre actes : un acte d'engagement, un acte de défi, un acte de réponse et un acte de contrôle. Le démonstrateur joue les actes d'engagement et de réponse. Le contrôleur joue les actes de défi et de contrôle.

Au sein du démonstrateur, on peut isoler un témoin, de manière à isoler les paramètres et les fonctions les plus sensibles du démonstrateur, c'est-à-dire, la production des engagements et des réponses. Le témoin dispose du paramètre k et de la clé privée GQ2, c'est-à-dire, de la factorisation du module n selon l'une des trois représentations évoquées ci-dessus : • les f facteurs premiers et les m nombres de base, • les $m \times f$ composantes privées, les f facteurs premiers et $f-1$ paramètres des restes chinois, • les m nombres privés et le module n.

Le témoin peut correspondre à une réalisation particulière, par exemple, • une carte à puce reliée à un PC formant ensemble le démonstrateur, ou encore, • des programmes particulièrement protégés au sein d'un PC, ou encore, • des programmes particulièrement protégés au sein d'une carte à puce. Le témoin ainsi isolé est semblable au témoin défini ci-après au sein du signataire. A chaque exécution du mécanisme, le témoin produit un ou plusieurs engagements R, puis, autant de réponses D à autant de défis d. Chaque ensemble $\{R, d, D\}$ constitue un **triplet GQ2**.

Outre qu'il comprend le témoin, le démonstrateur dispose également, le cas échéant, d'une fonction de hachage et d'un message M.

Le contrôleur dispose du module n, par exemple, à partir d'un annuaire de clés publiques ou encore à partir d'un certificat de clés publique ; le cas

échéant, il dispose également de la même fonction de hachage et d'un message M'. Les paramètres publics GQ2, à savoir les nombres k, m et g_1 à g_m peuvent être donnés au contrôleur par le démonstrateur. Le contrôleur est apte à reconstituer un engagement R' à partir de n'importe quel défi d et de n'importe quelle réponse D. Les paramètres k et m renseignent le contrôleur. Faute d'indication contraire, les m nombres de base de g_1 à g_m sont les m premiers nombres premiers. Chaque défi d doit comporter m défis élémentaires notés de d_1 à d_m: un par nombre de base. Chaque défi élémentaire de d_1 à d_m est un nombre de 0 à $2^{k-1}-1$ (les nombres de $v/2$ à $v-1$ ne sont pas utilisés). Typiquement, chaque défi est codé par m fois $k-1$ bits (et non pas m fois k bits). Par exemple, avec $k = 5$ et $m = 4$ nombres de base 5, 11, 21 et 26, chaque défi comporte 16 bits transmis sur quatre quartets. Lorsque les $(k-1)\times m$ défis possibles sont également probables, le nombre $(k-1)\times m$ détermine la sécurité apportée par chaque triplet GQ2 : un imposteur qui, par définition, ne connaît pas la factorisation du module n a exactement une chance de succès sur $2^{(k-1)\times m}$. Lorsque $(k-1)\times m$ vaut de 15 à 20, un triplet suffit à assurer raisonnablement l'authentification dynamique. Pour atteindre n'importe quel niveau de sécurité, on peut produire des triplets en parallèle ; on peut également en produire en séquence, c'est-à-dire, répéter l'exécution du mécanisme.

1) L'acte d'engagement comprend les opérations suivantes.

Lorsque le témoin n'utilise pas les restes chinois, il dispose du paramètre k, des m nombres privés de Q_1 à Q_m et du module n ; il tire au hasard et en privé un ou plusieurs aléas r ($0 < r < n$) ; puis, par k élévations successives au carré (mod n), il transforme chaque aléa r en un engagement R.

$$R \equiv r^v \pmod{n}$$

Voici un exemple avec le jeu de clés précédent sans les restes chinois.

r = 5E94B894AC24AF843131F437C1B1797EF562CFA53AB8AD426C1
AC016F1C89CFDA13120719477C3E2FB4B4566088E10EF9C010E8F09

C60D981512198126091996

R = 6BBF9FFA5D509778D0F93AE074D36A07D95FFC38F70C8D7E330 0EBF234FA0BC20A95152A8FB73DE81FAEE5BF4FD3EB7F5EE3E36D 7068D083EF7C93F6FDDF673A

Lorsque le témoin utilise les restes chinois, il dispose du paramètre k, des f facteurs premiers de p_1 à p_f, de $f-1$ paramètres des restes chinois et des $m \times f$ composantes privées $Q_{i,j}$; il tire au hasard et en privé une ou plusieurs collections de f aléas : chaque collection comporte un aléa r_i par facteur premier p_i $(0 < r_i < p_i)$; puis, par k élévations successives au carré (mod p_i), il transforme chaque aléa r_i en une composante d'engagement R_i.

$$R_i \equiv r_i^v \pmod{p_i}$$

Pour chaque collection de f composantes d'engagement, le témoin établit un engagement selon la technique des restes chinois. Il y a autant d'engagements que de collections d'aléas.

$$R = \text{Restes Chinois}(R_1, R_2, \ldots R_f)$$

Voici un exemple avec le jeu de clés précédent et avec les restes chinois.

r_1 = 5C6D37F0E97083C8D120719475E080BBBF9F7392F11F3E244FDF0 204E84D8CAE

R_1 = 3DDF516EE3945CB86D20D9C49E0DA4D42281D07A76074DD4FE C5C7C5E205DF66

r_2 = AC8F85034AC78112071947C457225E908E83A2621B0154ED15DB FCB9A4915AC3

R_2 = 01168CEC0F661EAA15157C2C287C6A5B34EE28F8EB4D8D34085 8079BCAE4ECB016

R = Restes Chinois(R_1, R_2) = 0AE51D90CB4FDC3DC757C56E063C9ED8 6BE153B71FC65F47C123C27F082BC3DD15273D4A923804718573F2F0 5E991487D17DAE0AAB7DF0D0FFA23E0FE59F95F0

Dans les deux cas, le démonstrateur transmet au contrôleur tout ou partie de chaque engagement $\boldsymbol{R}$, ou bien, un code de hachage $\boldsymbol{H}$ obtenu en hachant

chaque engagement $\boldsymbol{R}$ et un message $\boldsymbol{M}$.

2) L'acte de défi consiste à tirer au hasard un ou plusieurs défis d composés chacun de m défis élémentaires d_1 d_2 ... d_m ; chaque défi élémentaire d_i est l'un des nombres de 0 à $v/2-1$.

$$d = d_1\ d_2 \ldots\ d_m$$

Voici un défi pour les deux exemples, c'est-à-dire avec $k = 5$ et $m = 4$.

$$d_1 = 1011 = 11 = \text{‘B’}\ ;\ d_2 = 0011 = 3\ ;\ d_3 = 0110 = 6\ ;\ d_4 = 1001 = 9,$$

$$d = d_1 \,||\, d_2 \,||\, d_3 \,||\, d_4 = 10110011\ 01101001 = \text{B3 69}$$

Le contrôleur transmet au démonstrateur chaque défi $\boldsymbol{d}$.

3) L'acte de réponse comporte les opérations suivantes.

Lorsque le témoin n'utilise pas les restes chinois, il dispose du paramètre k, des m nombres privés de Q_1 à Q_m et du module n ; il calcule une ou plusieurs réponses D en utilisant chaque aléa r de l'acte d'engagement et les nombres privés selon les défis élémentaires.

$$D \equiv r \times Q_1^{d_1} \times Q_2^{d_2} \times \ldots Q_m^{d_m} \pmod n$$

Voici la suite de l'exemple sans les restes chinois.

$D =$ 027E6E808425BF2B401FD00B15B642B1A8453BE8070D86C0A787
0E6C1940F7A6996C2D871EBE611812532AC5875E0E116CC8BA648FD
8E86BE0B2ABCC3CCBBBE4

Lorsque le témoin utilise les restes chinois, il dispose du paramètre k, des f facteurs premiers de p_1 à p_f, de $f-1$ paramètres des restes chinois et des $m{\times}f$ composantes privées $Q_{i,j}$; il calcule une ou plusieurs collections de f composantes de réponse en utilisant chaque collection d'aléas de l'acte d'engagement : chaque collection de composantes de réponse comporte une composante par facteur premier.

$$D_i \equiv r_i \times Q_{1,i}^{d_1} \times Q_{2,i}^{d_2} \times \ldots Q_{m,i}^{d_m} \pmod{p_i}$$

Pour chaque collection de composantes de réponse, le témoin établit une réponse selon la technique des restes chinois. Il y a autant de réponses que de défis.

$$D = \text{Restes Chinois}(D_1, D_2, \ldots D_f)$$

Voici la suite de l'exemple avec les restes chinois.

$D_1 = r_1 \times Q_{1,1}{}^{d1} \times Q_{2,1}{}^{d2} \times Q_{3,1}{}^{d3} \times Q_{4,1}{}^{d4} \pmod{p_1} =$
C71F86F6FD8F955E2EE434BFA7706E38E5E715375BC2CD2029A4BD
572A9EDEE6

$D_2 = r_2 \times Q_{1,2}{}^{d1} \times Q_{2,2}{}^{d2} \times Q_{3,2}{}^{d3} \times Q_{4,2}{}^{d4} \pmod{p_2} =$
0BE022F4A20523F98E9F5DBEC0E10887902F3AA48C864A6C354693A
D0B59D85E

$D =$ 90CE7EA43CB8EA89ABDD0C814FB72ADE74F02FE6F098ABB98
C8577A660B9CFCEAECB93BE1BCC356811BF12DD667E2270134C907
3B9418CA5EBF5191218D3FDB3

Dans les deux cas, le démonstrateur transmet chaque réponse ***D*** au contrôleur.

4) L'acte de contrôle consiste à contrôler que chaque triplet $\{R, d, D\}$ vérifie une équation du type suivant pour une valeur non nulle,

$$R \times \prod_{i=1}^{m} G_i^{d_i} \equiv D^{2^k} \pmod{n} \quad \text{ou bien} \quad R \equiv D^{2^k} \times \prod_{i=1}^{m} G_i^{d_i} \pmod{n}$$

ou bien, à rétablir chaque engagement : aucun ne doit être nul.

$$R' \equiv D^{2^k} / \prod_{i=1}^{m} G_i^{d_i} \pmod{n} \quad \text{ou bien} \quad R' \equiv D^{2^k} \times \prod_{i=1}^{m} G_i^{d_i} \pmod{n}$$

Eventuellement, le contrôleur calcule ensuite un code de hachage H' en hachant chaque engagement rétabli R' et un message M'. L'authentification dynamique est réussie lorsque le contrôleur retrouve ainsi ce qu'il a reçu à l'issue de l'acte d'engagement, c'est-à-dire, tout ou partie de chaque engagement R, ou bien, le code de hachage H.

Par exemple, une séquence d'opérations élémentaires transforme la réponse D en un engagement R'. La séquence comprend k carrés (mod n) séparés par $k-1$ divisions ou multiplications (mod n) par des nombres de base. Pour la i ième division ou multiplication, qui s'effectue entre le i ième carré et le

i+1 ième carré, le i ième bit du défi élémentaire d_1 indique s'il faut utiliser g_1, le i ième bit du défi élémentaire d_2 indique s'il faut utiliser g_2, ... jusqu'au i ième bit du défi élémentaire d_m qui indique s'il faut utiliser g_m.

Voici la fin de l'exemple sans les restes chinois.

D = 027E6E808425BF2B401FD00B15B642B1A8453BE8070D86C0A787
0E6C1940F7A6996C2D871EBE611812532AC5875E0E116CC8BA648FD
8E86BE0B2ABCC3CCBBBE4

Elever au carré modulo n :

88BA681DD641D37D7A7D9818D0DBEA82174073997C6C32F7FCAB3
0380C4C6229B0706D1AF6EBD84617771C31B4243C2F0376CAF5DCE
B644F098FAF3B1EB49B39

Multiplier par 5 fois 26 = 130, soit '82' modulo n :

6ECABA65A91C22431C413E4EC7C7B39FDE14C9782C94FD6FA3CA
AD7AFE192B9440C1113CB8DBC45619595D263C1067D3D0A840FDE0
08B415028AB3520A6AD49D

Elever au carré modulo n :

0236D25049A5217B13818B39AFB009E4D7D52B17486EBF844D64CF7
5C4F652031041328B29EBF0829D54E3BD17DAD218174A01E6E3AA65
0C6FD62CC274426607

Multiplier par 21, soit '15' modulo n :

2E7F40960A8BBF1899A06BBB6970CFC5B47C88E8F115B5DA594504
A92834BA405559256A705ABAB6E7F6AE82F4F33BF9E91227F0ACFA
4A052C91ABF389725E93

Elever au carré modulo n :

B802171179648AD687E672D3A32640E2493BA2E82D5DC87DBA2B2C
C0325E7A71C50E8AE02E299EF868DD3FB916EBCBC0C5569B53D42
DAD49C956D8572E1285B0

Multiplier par 5 fois 11 fois 21 = 1155, soit '483' modulo n :

3305560276310DEFEC1337EB5BB5810336FDB28E91B350D485B09188

E0C4F1D67E68E9590DB7F9F39C22BDB4533013625011248A8DC417C 667B419D27CB11F72

Elever au carré modulo *n* :

8871C494081ABD1AEB8656C38B9BAAB57DBA72A4BD4EF9029ECB FFF540E55138C9F22923963151FD0753145DF70CE22E9D019990E41D B6104005EEB7B1170559

Multiplier par 5 fois 11 fois 26 = 1430, soit '596' modulo *n* :

2CF5F76EEBF128A0701B56F837FF68F81A6A5D175D0AD67A14DAE C6FB68C362B1DC0ADD6CFC004FF5EEACDF794563BB09A17045EC FFF88F5136C7FBC825BC50C

Elever au carré modulo *n* :

6BBF9FFA5D509778D0F93AE074D36A07D95FFC38F70C8D7E3300EB F234FA0BC20A95152A8FB73DE81FAEE5BF4FD3EB7F5EE3E36D706 8D083EF7C93F6FDDF673A

On retrouve bien l'engagement ***R***. L'authentification est réussie.

Voici la fin de l'exemple avec les restes chinois.

D = 90CE7EA43CB8EA89ABDD0C814FB72ADE74F02FE6F098ABB98 C8577A660B9CFCEAECB93BE1BCC356811BF12DD667E2270134C907 3B9418CA5EBF5191218D3FDB3

Elever au carré modulo *n* :

770192532E9CED554A8690B88F16D013010C903172B266C1133B136E BE3EB5F13B170DD41F4ABE14736ADD3A70DFA43121B6FC5560CD D4B4845395763C792A68

Multiplier par 5 fois 26 = 130, soit '82' modulo *n* :

6EE9BEF9E52713004971ABB9FBC31145318E2A703C8A2FB3E144E77 86397CD8D1910E70FA86262DB771AD1565303AD6E4CC6E90AE3646 B461D3521420E240FD4

Elever au carré modulo *n* :

D9840D9A8E80002C4D0329FF97D7AD163D8FA98F6AF8FE2B2160B2

126CBBDFC734E39F2C9A39983A426486BC477F20ED2CA59E664C23
CA0E04E84F2F0AD65340

Multiplier par 21, soit '15' modulo n :

D7DD7516383F78944F2C90116E1BEE0CCDC8D7CEC5D7D1795ED33
BFE8623DB3D2E5B6C5F62A56A2DF4845A94F32BF3CAC360C7782B
5941924BB4BE91F86BD85F

Elever au carré modulo n :

DD34020DD0804C0757F29A0CBBD7B46A1BAF949214F74FDFE021B
626ADAFBAB5C3F1602095DA39D70270938AE362F2DAE0B91485531
0C7BCA328A4B2643DCCDF

Multiplier par 5 fois 11 fois 21 = 1155, soit '483' modulo n :

038EF55B4C826D189C6A48EFDD9DADBD2B63A7D675A0587C85596
18EA2D83DF552D24EAF6BE983FB4AFB3DE7D4D2545190F1B1F946
D327A4E9CA258C73A98F57

Elever au carré modulo n :

D1232F50E30BC6B7365CC2712E5CAE079E47B971DA03185B33E918E
E6E99252DB3573CC87C604B327E5B20C7AB920FDF142A8909DBBA1
C04A6227FF18241C9FE

Multiplier par 5 fois 11 fois 26 = 1430, soit '596' modulo n :

3CC768F12AEDFCD4662892B9174A21D1F0DD9127A54AB63C984019
BED9BF88247EF4CCB56D71E0FA30CFB0FF28B7CE45556F744C1FD7
51BFBCA040DC9CBAB744

Elever au carré modulo n :

0AE51D90CB4FDC3DC757C56E063C9ED86BE153B71FC65F47C123C
27F082BC3DD15273D4A923804718573F2F05E991487D17DAE0AAB7
DF0D0FFA23E0FE59F95F0

On retrouve bien l'engagement R. L'authentification est réussie.

Signature numérique

Le mécanisme de signature numérique permet à une entité appelée

signataire de produire des messages signés et à une entité appelée **contrôleur** de vérifier des messages signés. Le message M est une séquence binaire quelconque : il peut être vide. Le message M est signé en lui adjoignant un appendice de signature qui comprend un ou plusieurs engagements et / ou défis, ainsi que les réponses correspondantes.

Le contrôleur dispose du module n, par exemple, à partir d'un annuaire de clés publiques ou encore à partir d'un certificat de clés publique ; il dispose également de la même fonction de hachage. Les paramètres publics GQ2, à savoir les nombres k, m et g_1 à g_m peuvent être donnés au contrôleur par le démonstrateur, par exemple, en les mettant dans l'appendice de signature.

Les nombres k et m renseignent le contrôleur. D'une part, chaque défi élémentaire, de d_1 à d_m, est un nombre de 0 à $2^{k-1}-1$ (les nombres $v/2$ à $v-1$ ne sont pas utilisés). D'autre part, chaque défi d doit comporter m défis élémentaires notés de d_1 à d_m, autant que de nombres de base. En outre, faute d'indication contraire, les m nombres de base, de g_1 à g_m, sont les m premiers nombres premiers. Avec $(k-1)\times m$ valant de 15 à 20, on peut signer avec quatre triplets GQ2 produits en parallèle ; avec $(k-1)\times m$ valant 60 ou plus, on peut signer avec un seul triplet GQ2. Par exemple, avec $k = 9$ et $m = 8$, un seul triplet GQ2 suffit ; chaque défi comporte huit octets et les nombres de base sont 2, 3, 5, 7, 11, 13, 17 et 19.

L'opération de signature est une séquence de trois actes : un acte d'engagement, un acte de défi et un acte de réponse. Chaque acte produit un ou plusieurs triplets GQ2 comprenant chacun : un engagement R ($\neq 0$), un défi d composé de m défis élémentaires notés par d_1, d_2, ... d_m et une réponse D ($\neq 0$).

Le signataire dispose d'une fonction de hachage, du paramètre k et de la clé privée GQ2, c'est-à-dire, de la factorisation du module n selon l'une des trois représentations évoquées ci-dessus. **Au sein du signataire, on peut isoler un témoin qui exécute les actes d'engagement et de réponse,** de

manière à isoler les fonctions et les paramètres les plus sensibles du démonstrateur. Pour calculer engagements et réponses, le témoin dispose du paramètre k et de la clé privée GQ2, c'est-à-dire, de la factorisation du module n selon l'une des trois représentations évoquées ci-dessus. Le témoin ainsi isolé est semblable au témoin défini au sein du démonstrateur. Il peut correspondre à une réalisation particulière, par exemple, • une carte à puce reliée à un PC formant ensemble le signataire, ou encore, • des programmes particulièrement protégés au sein d'un PC, ou encore, • des programmes particulièrement protégés au sein d'une carte à puce.

1) L'acte d'engagement comprend les opérations suivantes.

Lorsque le témoin dispose des m nombres privés Q_1 à Q_m et du module n, il tire au hasard et en privé un ou plusieurs aléas r ($0 < r < n$) ; puis, par k élévations successives au carré (mod n), il transforme chaque aléa r en un engagement R.

$$R \equiv r^v \pmod n$$

Lorsque le témoin dispose des f facteurs premiers de p_1 à p_f et des $m\times f$ composantes privées $Q_{i,j}$, il tire au hasard et en privé une ou plusieurs collections de f aléas : chaque collection comporte un aléa r_i par facteur premier p_i ($0 < r_i < p_i$) ; puis, par k élévations successives au carré (mod p_i), il transforme chaque aléa r_i en une composante d'engagement R_i.

$$R_i \equiv r_i^v \pmod{p_i}$$

Pour chaque collection de f composantes d'engagement, le témoin établit un engagement selon la technique des restes chinois. Il y a autant d'engagements que de collections d'aléas.

$$R = \text{Restes Chinois}(R_1, R_2, \ldots R_f)$$

2) L'acte de défi consiste à hacher tous les engagements R et le message à signer M pour obtenir un code de hachage à partir duquel le signataire forme un ou plusieurs défis comprenant chacun m défis élémentaires ; chaque défi élémentaire est un nombre de 0 à $v/2-1$; par exemple, avec

$k = 9$ et $m = 8$, chaque défi comporte huit octets. Il y a autant de défis que d'engagements.

$$d = d_1 \; d_2 \; \ldots \; d_m, \text{ extraits du résultat Hash}(M, R)$$

3) L'acte de réponse comporte les opérations suivantes.

Lorsque la témoin dispose des m nombres privés Q_1 à Q_m et du module n, il calcule une ou plusieurs réponses D en utilisant chaque aléa r de l'acte d'engagement et les nombres privés selon les défis élémentaires.

$$X \equiv Q_1^{d_1} \times Q_2^{d_2} \times \ldots Q_m^{d_m} \pmod n$$

$$D \equiv r \times X \pmod n$$

Lorsque le témoin dispose des f facteurs premiers de p_1 à p_f et des $m \times f$ composantes privées $Q_{i,j}$, il calcule une ou plusieurs collections de f composantes de réponse en utilisant chaque collection d'aléas de l'acte d'engagement : chaque collection de composantes de réponse comporte une composante par facteur premier.

$$X_i \equiv Q_{1,i}^{d_1} \times Q_{2,i}^{d_2} \times \ldots Q_{m,i}^{d_m} \pmod{p_i}$$

$$D_i \equiv r_i \times X_i \pmod{p_i}$$

Pour chaque collection de composantes de réponse, le témoin établit une réponse selon la technique des restes chinois. Il y a autant de réponses que de défis.

$$D = \text{Restes Chinois}(D_1, D_2, \ldots D_f)$$

Le signataire signe le message M en lui adjoignant un appendice de signature comprenant :

- ou bien, chaque triplet GQ2, c'est-à-dire, chaque engagement R, chaque défi d et chaque réponse D,
- ou bien, chaque engagement R et chaque réponse D correspondante,
- ou bien, chaque défi d et chaque réponse D correspondante.

Le déroulement de l'opération de vérification dépend du contenu de l'appendice de signature. On distingue les trois cas.

Au cas où l'appendice comprend un ou plusieurs triplets, l'opération de

contrôle comporte deux processus indépendants dont la chronologie est indifférente. Le contrôleur accepte le message signé si et seulement si les deux conditions suivantes sont remplies.

D'une part, chaque triplet doit être cohérent (une relation appropriée du type suivant doit être vérifiée) et recevable (la comparaison doit se faire sur une valeur non nulle).

$$R \times \prod_{i=1}^{m} G_i^{d_i} \equiv D^{2^k} \pmod{n} \quad \text{ou bien} \quad R \equiv D^{2^k} \times \prod_{i=1}^{m} G_i^{d_i} \pmod{n}$$

Par exemple, on transforme la réponse D par une séquence d'opérations élémentaires : k carrés (mod n) séparés par $k-1$ multiplications ou divisions (mod n) par des nombres de base. Pour la i ième multiplication ou division, qui s'effectue entre le i ième carré et le $i+1$ ième carré, le i ième bit du défi élémentaire d_1 indique s'il faut utiliser g_1, le i ième bit du défi élémentaire d_2 indique s'il faut utiliser g_2, ... jusqu'au i ième bit du défi élémentaire d_m qui indique s'il faut utiliser g_m. On doit ainsi retrouver chaque engagement R présent dans l'appendice de signature.

D'autre part, le ou les triplets doivent être liés au message M. En hachant tous les engagements R et le message M, on obtient un code de hachage à partir duquel on doit retrouver chaque défi d.

$d = d_1\ d_2 \dots\ d_m$, identiques à ceux extraits du résultat Hash(M, R)

Au cas où l'appendice ne comprend pas de défi, l'opération de contrôle commence par la reconstitution de un ou plusieurs défis d' en hachant tous les engagements R et le message M.

$d' = d'_1\ d'_2 \dots\ d'_m$, extraits du résultat Hash(M, R)

Ensuite, le contrôleur accepte le message signé si et seulement si chaque triplet est cohérent (une relation appropriée du type suivant est vérifiée) et recevable (la comparaison se fait sur une valeur non nulle).

$$R \times \prod_{i=1}^{m} G_i^{d'_i} \equiv D^{2^k} \pmod{n} \quad \text{ou bien} \quad R \equiv D^{2^k} \times \prod_{i=1}^{m} G_i^{d'_i} \pmod{n}$$

Au cas où l'appendice ne comprend pas d'engagement, l'opération de contrôle commence par la reconstitution de un ou plusieurs engagements *R'* selon une des deux formules suivantes, celle qui est appropriée. Aucun engagement rétabli ne doit être nul.

$$R' \equiv D^{2^k} / \prod_{i=1}^{m} G_i^{d_i} \pmod{n} \quad \text{ou bien} \quad R' \equiv D^{2^k} \times \prod_{i=1}^{m} G_i^{d_i} \pmod{n}$$

Ensuite, le contrôleur doit hacher tous les engagements *R'* et le message *M* de façon à reconstituer chaque défis *d*.

$d = d_1\ d_2 \dots\ d_m$, identiques à ceux extraits du résultat Hash(*M*, *R'*)

Le contrôleur accepte le message signé si et seulement si chaque défi reconstitué est identique au défi correspondant figurant en appendice.

Revendications

1. Dans un procédé destiné à prouver à une entité contrôleur,

- l'authenticité d'une entité et/ou
- l'intégrité d'un message **M** associé à cette entité ;

ledit procédé mettant en œuvre :

- un module public **n** constitué par le produit de **f** facteurs premiers p_1, p_2, ... p_f (**f** étant supérieur ou égal à 2) ou mettant en œuvre les **f** facteurs premiers ;
- **m** nombres de base entiers g_1, g_2, ... g_m distincts (**m** étant supérieur ou égal à 1), g_i étant inférieur aux **f** facteurs premiers p_1, p_2, ... p_f ;
- **m** couples de valeurs privées Q_1, Q_2, ... Q_m et publiques G_1, G_2, ... G_m (**m** étant supérieur ou égal à 1) ou des paramètres dérivés de ceux-ci ;

ledit module et lesdites valeurs privées et publiques étant liés par des relations du type :

$$G_i \cdot Q_i^v \equiv 1 \bmod n \text{ ou } G_i \equiv Q_i^v \bmod n$$

ladite valeur publique G_i étant le carré g_i^2 du nombre de base,

v désignant un exposant public de la forme :

$$v = 2^k$$

où **k** est un paramètre de sécurité plus grand que 1 ;

le procédé selon l'invention comprenant l'étape de produire les **f** facteurs premiers p_1, p_2, ... p_f et/ou les **m** nombres de base g_1, g_2, ... g_m de telle sorte que les conditions suivantes soient satisfaites :

Première condition :

chacune des équations :

$$x^v \equiv g_i^2 \bmod n \qquad (1)$$

a des solutions en x dans l'anneau des entiers modulo **n** ;

Deuxième condition :

dans le cas où $G_i \equiv Q_i^v \bmod n$, parmi les **m** nombres q_i obtenus en élevant Q_i au carré modulo n, k-1 fois de rang, l'un d'entre eux est différent de $\pm g_i$

(c'est-à-dire est non trivial),
dans le cas où $G_i . Q_i^v \equiv 1 \bmod n$, parmi les **m** nombres q_i obtenus en élevant l'inverse de Q_i modulo n au carré modulo n, k-1 fois de rang, l'un d'entre eux est différent de $\pm g_i$ (c'est-à-dire est non trivial) ;

Troisième condition :

parmi les **2m** équations :

$$x^2 \equiv g_i \bmod n \qquad (2)$$

$$x^2 \equiv - g_i \bmod n \qquad (3)$$

au moins l'une d'entre elles a des solutions en x dans l'anneau des entiers modulo **n** ;

le procédé selon l'invention pour produire les **f** facteurs premiers $p_1, p_2, \dots p_f$, et/ou les **m** nombres de base $g_1, g_2, \dots g_m$ comprenant l'étape de choisir en premier :

- le paramètre de sécurité **k**,
- les **m** nombres de base $g_1, g_2, \dots g_m$, et/ou les **f** facteurs premiers $p_1, p_2, \dots p_f$

2. Procédé selon la revendication 1 tel que les **m** nombres de base $g_1, g_2, \dots g_m$, sont choisis au moins en partie parmi les premiers nombres entiers.

3. Procédé selon l'une des revendications 1 ou 2, tel que le paramètre de sécurité **k** est un petit nombre entier, notamment inférieur à 100.

4. Procédé selon l'une quelconque des revendications 1 à 3, tel que la taille du module **n** est supérieure à plusieurs centaines de bits.

5. Procédé selon l'une quelconque des revendications 1 à 4, tel que les **f** facteurs premiers $p_1, p_2, \dots p_f$ ont une taille voisine de la taille du module **n** divisé par le nombre **f** de facteurs.

6. Procédé selon l'une quelconques des revendications 1 à 5 tel que pour tester la première condition, on vérifie la compatibilité des nombres **k**, **p**, **g** en mettant en œuvre l'algorithme ci-après :

- on désigne par **h** un nombre tel que 2^h divise le rang de **g** par rapport

à **p** et tel que 2^{h+1} ne le divise pas,

- on calcule **h** à partir du symbole de Legendre (**g** | **p**) et d'un nombre **b** égal à une racine 2^t- ième primitive de l'unité dans **CG(p)**,

• si (g | p) = -1 alors h= t,

• si (g | p) = +1 avec t = 1, alors h= 0,

• si (g | p) = +1 avec t > 1, alors on applique la clé $\langle(p-1+2^t)/2^{t+1}$, $p\rangle$ à G, on obtient ainsi un résultat w :

• • si w = +g, alors h = 0,

• • si w = p-g, alors h = +1,

• • sinon, on met en œuvre le sous-module de calcul ci-après, en initialisant la variable c en lui attribuant la valeur b, puis en itérant les étapes suivantes pour des valeurs de i allant de t-1 à 2 :

étape 1 : on applique la clé $\langle 2^i, p\rangle$ à w/g(modp),

* si le résultat obtenu est égal à +1, on passe à l'étape 2,

* si le résultat obtenu est égal à -1, on attribue à h la valeur i et on remplace w par w.c(modp),

étape 2 : on remplace c par c^2 (modp),

la valeur de h recherchée est celle obtenue la dernière fois où l'application de la clé $\langle 2^i, p\rangle$, conformément à l'étape 1, a produit un résultat égal à –1,

(on rappelle que

- k, g, p sont incompatibles lorsque h>1 et lorsque k+h >t+1,

- k, g, p sont compatibles lorsque h = 0 ou 1, quelle que soit la valeur de k, ou lorsque h>1 et lorsque k+h ≤ t+1).

(dans ledit algorithme, le symbole de Legendre et t ont le sens défini dans la description).

7. Procédé selon la revendication 6 tel que pour tester la deuxième condition, on vérifie qu'au moins un jeu $\{\delta_{i,1} \ldots \delta_{i,f}\}$ est variable ou nul, *(δ à le sens défini dans la description).*

8. Procédé selon la revendication 7 tel que pour tester la troisième

condition, on vérifie qu'il existe un nombre de base g_i de g_1 à g_m tel que les f symboles de Legendre $(g_i \mid p_1)$ à $(g_i \mid p_f)$ soient tous égaux à +1 ou bien les f symboles de Legendre $(-g_i \mid p_1)$ à $(-g_i \mid p_f)$ soient tous égaux à +1.

9. Procédé selon l'une quelconques des revendications 1 à 8 tel que pour calculer les **f.m** composantes privées $\mathbf{Q_{i,j}}$ des valeurs privées $\mathbf{Q_1}$, $\mathbf{Q_2}$, ... $\mathbf{Q_m}$ ($\mathbf{Q_{i,j} \equiv Q_i \bmod p_j}$), dans le cas où $\mathbf{G_i \equiv Q_i^v \bmod n}$:

- si $t = 1$ (c'est-à-dire si $p_j \equiv 3 \pmod 4$) :
  - on calcule un nombre s_j tel que $s_j \equiv ((p_j + 1)/4)^k \pmod{(p_j - 1)/2}$,
  - on en déduit la clé $\langle s_j, p_j \rangle$,
  - on applique la clé $\langle s_j, p_j \rangle$ à $\mathbf{G_i}$,
  - on obtient ainsi : $w \equiv \mathbf{G_i}^{sj} \pmod{p_j}$,
  - les deux valeurs possibles de $\mathbf{Q_{i,j}}$ sont w, $p_j - w$,
- si $t = 2$ (c'est-à-dire si $p_j \equiv 5 \pmod 8$) :
  - on calcule un nombre s_j tel que $s_j \equiv ((p_j + 3)/8)^k \pmod{(p_j - 1)/4}$,
  - on en déduit la clé $\langle s_j, p_j \rangle$,
  - on applique la clé $\langle s_j, p_j \rangle$ à $\mathbf{G_i}$,
  - on obtient ainsi : $w \equiv \mathbf{G_i}^{sj} \pmod{p_j}$ et $w' \equiv w.z \pmod{p_j}$,
  - les quatre valeurs possibles de $\mathbf{Q_{i,j}}$ sont w, $p_j - w$, w', $p_j - w'$,

(dans ledit algorithme z a le sens défini dans la description).

- si $t > 2$ (c'est-à-dire si $p_j \equiv 2^t + 1 \pmod{2^{t+1}}$) et si $h=0$ ou si $h = 1$,
  - on calcule s_j tel que $s_j \equiv ((p_j - 1 + 2^t)/ 2^{t+1})^k \pmod{(p_j - 1)/ 2^t}$,
  - on en déduit la clé $\langle s_j, p_j \rangle$,
  - on applique la clé $\langle s_j, p_j \rangle$ à $\mathbf{G_i}$,
  - on obtient ainsi : $w \equiv \mathbf{G_i}^{sj} \pmod{p_j}$,
  - les $2^{\min(k,t)}$ valeurs possibles de $\mathbf{Q_{i,j}}$ sont égales au produit de w par l'une quelconque des racines $2^{\min(k,t)}$-ièmes de l'unité dans $CG(p_j)$,
- si $t > 2$ (c'est-à-dire si $p_j \equiv 2^t + 1 \pmod{2^{t+1}}$) et si $h > 1$ et si $h+k \leq t+1$,
  - on calcule s_j tel que $s_j \equiv ((p_j - 1 + 2^t)/ 2^{t+1})^{k+h-1} \pmod{(p_j - 1)/ 2^t}$,
  - on en déduit la clé $\langle s_j, p_j \rangle$,

• on applique la clé $\langle s_j, p_j \rangle$ à la puissance 2^{h-1}–ième de $\mathbf{G_i}$,

• on obtient ainsi w,

• les 2^k valeurs possibles de $\mathbf{Q_{i,j}}$ appartiennent à l'ensemble des produits de w par les racines 2^{k+h-1}–ièmes primitives de l'unité dans $CG(p_j)$.

10. Procédé selon la revendication 9 tel que pour calculer les composantes privées $\mathbf{Q_{i,j}}$ dans le cas où $\mathbf{G_i . Q_i^v \equiv 1 . mod\ n}$, on substitue s_j par $((p_j - 1)/2^t) - s_j$ dans la clé $\langle s_j, p_j \rangle$.

11. Procédé faisant application du procédé, selon l'une quelconque des revendications 1 à 8, permettant de produire les **f** facteurs premiers $\mathbf{p_1}$, $\mathbf{p_2}$, ... $\mathbf{p_f}$ ou les **m** nombres de base $\mathbf{g_1}$, $\mathbf{g_2}$, ... $\mathbf{g_m}$;
ledit procédé étant destiné à prouver à une entité contrôleur

- l'authenticité d'une entité et/ou

- l'intégrité d'un message **M** associé à cette entité,

au moyen de **m** couples de valeurs privées $\mathbf{Q_1}$, $\mathbf{Q_2}$, ... $\mathbf{Q_m}$ et publiques $\mathbf{G_1}$, $\mathbf{G_2}$, ... $\mathbf{G_m}$ (**m** étant supérieur ou égal à 1) ou des paramètres dérivés de ceux-ci, notamment au moyen des composantes privées $\mathbf{Q_{i,j}}$;
ledit procédé mettant en œuvre selon les étapes ci-après une entité appelée témoin ;
ladite entité témoin disposant des **f** facteurs premiers $\mathbf{p_i}$ et/ou des paramètres des restes chinois des facteurs premiers ; et/ou du module public **n** et/ou des **m** valeurs privées $\mathbf{Q_i}$ et/ou des **f.m** composantes privées $\mathbf{Q_{i,j}}$ des valeurs privées $\mathbf{Q_i}$ et de l'exposant public **v** ;

- le témoin calcule des engagements **R** dans l'anneau des entiers modulo **n** ; chaque engagement étant calculé :

• soit en effectuant des opérations du type

$$\mathbf{R \equiv r^v \bmod n}$$

où **r** est un aléa tel que $\mathbf{0 < r < n}$,

• soit

•• en effectuant des opérations du type

$$\mathbf{R_i \equiv r_i^v \bmod p_i}$$

où $\mathbf{r_i}$ est un aléa associé au nombre premier $\mathbf{p_i}$ tel que $\mathbf{0 < r_i < p_i}$, chaque $\mathbf{r_i}$ appartenant à une collection d'aléas $\{\mathbf{r_1, r_2, \ldots r_f}\}$,

•• puis en appliquant la méthode des restes chinois ;

- le témoin reçoit un ou plusieurs défis **d** ; chaque défi **d** comportant **m** entiers $\mathbf{d_i}$ ci-après appelés défis élémentaires ; le témoin calcule à partir de chaque défi **d** une réponse **D**,

• soit en effectuant des opérations du type :

$$\mathbf{D \equiv r \,.\, Q_1^{\,d1} \,.\, Q_2^{\,d2} \,.\, \ldots\, Q_m^{\,dm} \bmod n}$$

• soit

•• en effectuant des opérations du type :

$$\mathbf{D_i \equiv r_i \,.\, Q_{i,1}^{\,d1} \,.\, Q_{i,2}^{\,d2} \,.\, \ldots\, Q_{i,m}^{\,dm} \bmod p_i}$$

•• puis en appliquant la méthode des restes chinois ;

ledit procédé étant tel qu'il y a autant de réponses **D** que de défis **d** et d'engagements **R**, chaque groupe de nombres **R, d, D** constituant un triplet noté **{R, d, D}**.

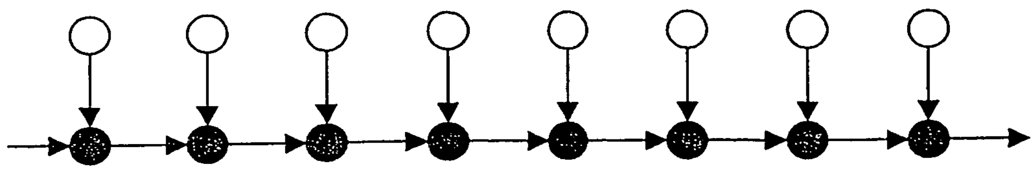

Fig.1A

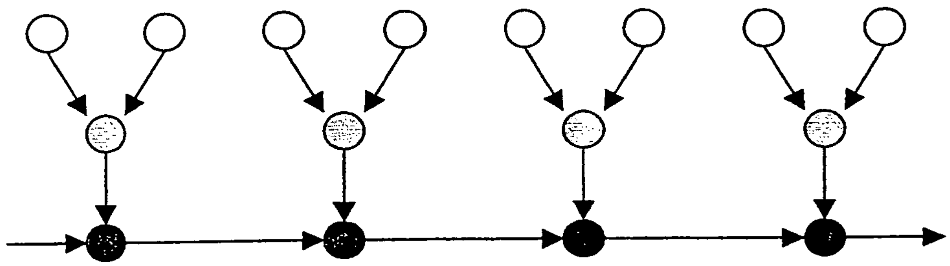

Fig.1B

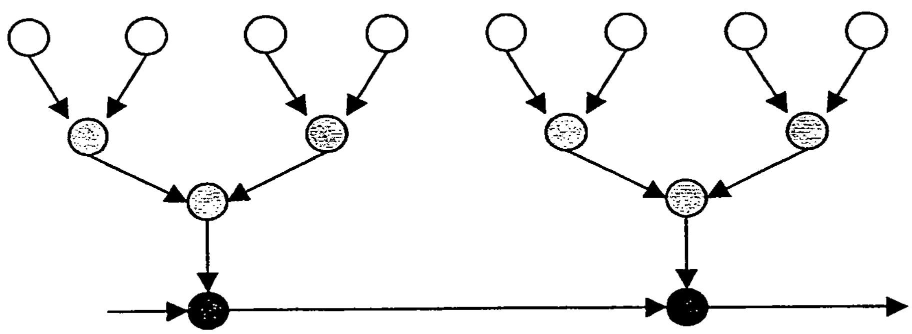

Fig.1C

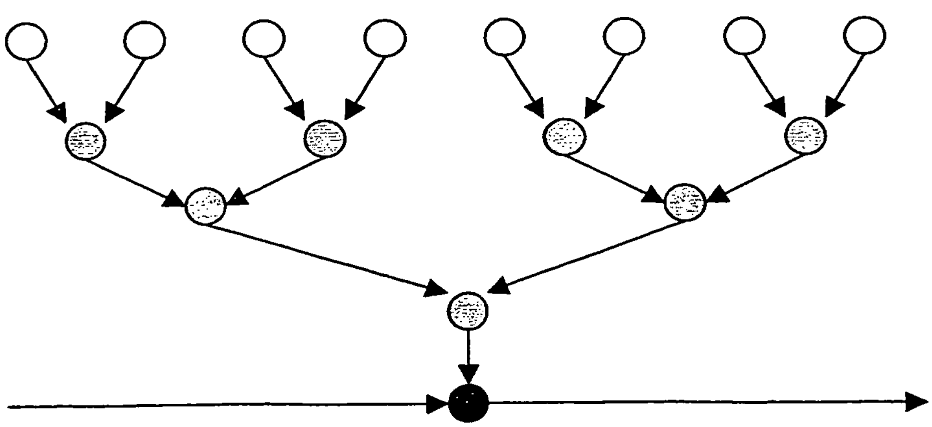

Fig.1D

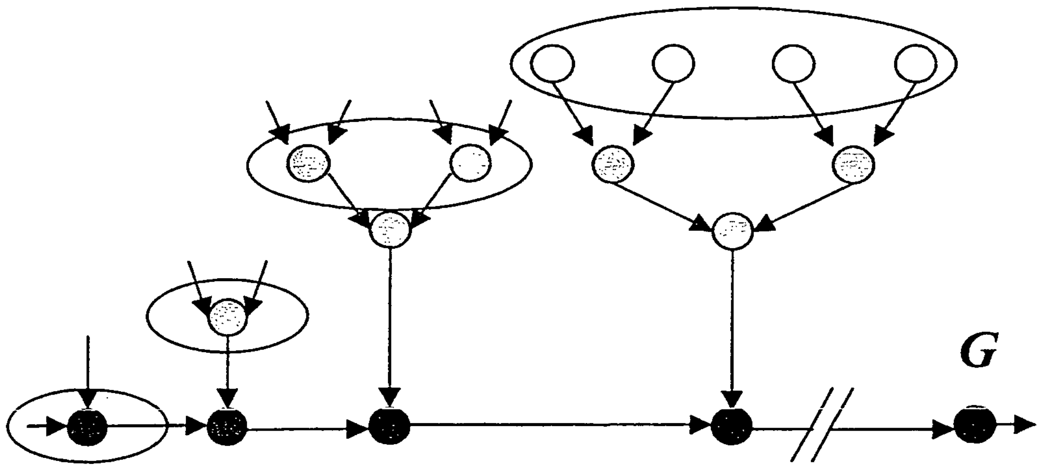

Fig.2A

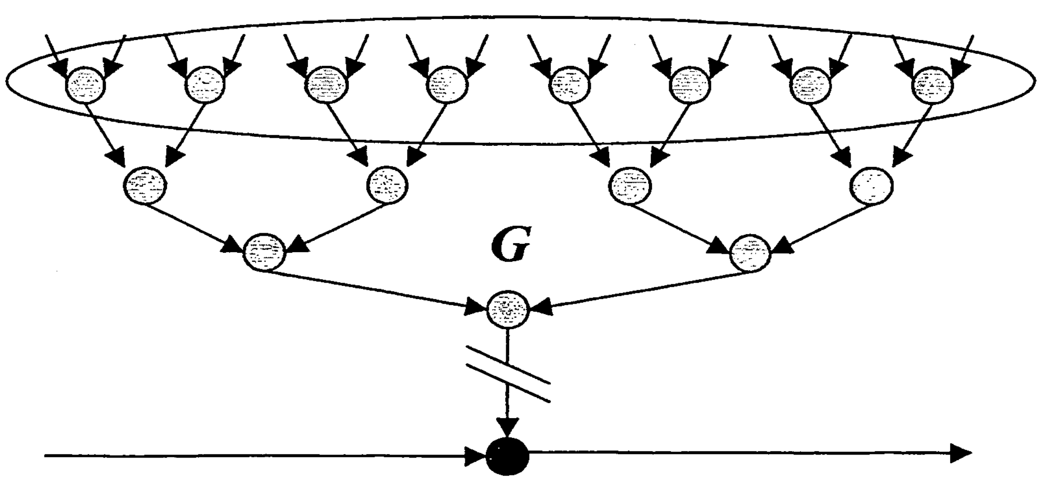

Fig.2B

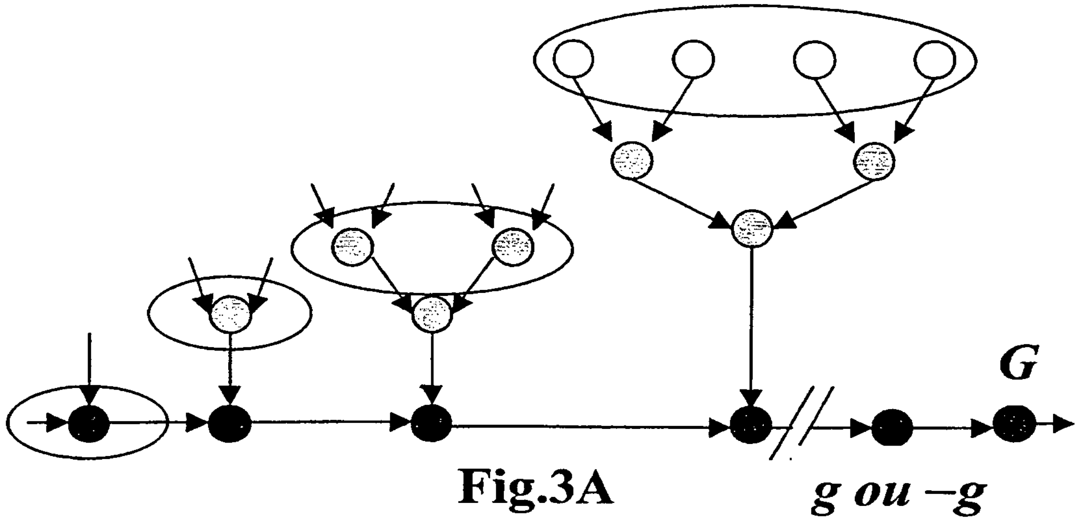

Fig.3A

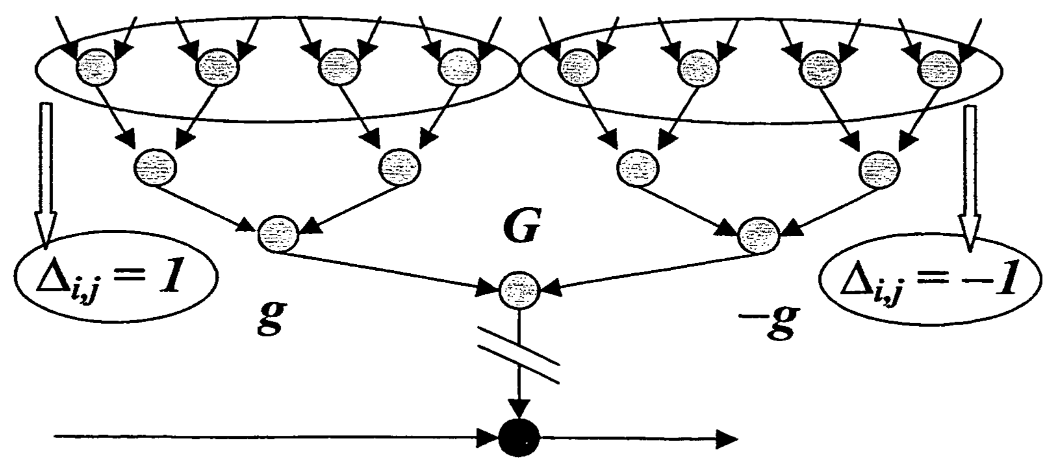

Fig.3B

RAPPORT DE RECHERCHE INTERNATIONALE

Demande internationale No
PCT/FR 00/02715

A. CLASSEMENT DE L'OBJET DE LA DEMANDE
CIB 7 H04L9/32

Selon la classification internationale des brevets (CIB) ou à la fois selon la classification nationale et la CIB

B. DOMAINES SUR LESQUELS LA RECHERCHE A PORTE

Documentation minimale consultee (système de classification suivi des symboles de classement)
CIB 7 H04L

Documentation consultée autre que la documentation minimale dans la mesure où ces documents relèvent des domaines sur lesquels a porté la recherche

Base de données électronique consultée au cours de la recherche internationale (nom de la base de données, et si réalisable, termes de recherche utilisés)

EPO-Internal, WPI Data, PAJ, INSPEC

C. DOCUMENTS CONSIDERES COMME PERTINENTS

Catégorie °	Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents	no. des revendications visées
A	EP 0 381 523 A (TOKYO SHIBAURA ELECTRIC CO) 8 août 1990 (1990-08-08) page 2, ligne 25 -page 3, ligne 7 ---	1,3,4
A	EP 0 311 470 A (TELEDIFFUSION FSE ;FRANCE ETAT (FR); PHILIPS NV (NL)) 12 avril 1989 (1989-04-12) cité dans la demande abrégé colonne 2, ligne 40 -colonne 3, ligne 50 colonne 12, ligne 30 -colonne 13, ligne 55 --- -/--	1,5,11

[X] Voir la suite du cadre C pour la fin de la liste des documents

[X] Les documents de familles de brevets sont indiqués en annexe

° Catégories spéciales de documents cités:

"A" document définissant l'état général de la technique, non considéré comme particulièrement pertinent

"E" document antérieur, mais publié à la date de dépôt international ou après cette date

"L" document pouvant jeter un doute sur une revendication de priorité ou cité pour déterminer la date de publication d'une autre citation ou pour une raison spéciale (telle qu'indiquée)

"O" document se référant à une divulgation orale, à un usage, à une exposition ou tous autres moyens

"P" document publié avant la date de dépôt international, mais postérieurement à la date de priorité revendiquée

"T" document ultérieur publié après la date de dépôt international ou la date de priorité et n'appartenenant pas à l'état de la technique pertinent, mais cité pour comprendre le principe ou la théorie constituant la base de l'invention

"X" document particulièrement pertinent; l'inven tion revendiquée ne peut être considérée comme nouvelle ou comme impliquant une activité inventive par rapport au document considéré isolément

"Y" document particulièrement pertinent; l'inven tion revendiquée ne peut être considérée comme impliquant une activité inventive lorsque le document est associé à un ou plusieurs autres documents de même nature, cette combinaison étant évidente pour une personne du métier

"&" document qui fait partie de la même famille de brevets

Date à laquelle la recherche internationale a été effectivement achevée	Date d'expédition du présent rapport de recherche internationale
14 décembre 2000	29/12/2000

Nom et adresse postale de l'administration chargée de la recherche internationale	Fonctionnaire autorisé
Office Européen des Brevets, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Masche, C

1

Deman ternationale No
PCT/FR 00/02715

C.(suite) DOCUMENTS CONSIDERES COMME PERTINENTS		
Catégorie °	Identification des documents cités, avec,le cas échéant, l'indicationdes passages pertinents	no. des revendications visées
A	QUISQUATER J -J ET AL: "FAST DECIPHERMENT ALGORITHM FOR RSA PUBLIC-KEY CRYPTOSYSTEM" ELECTRONICS LETTERS,IEE STEVENAGE,GB, vol. 18, no. 21, 14 octobre 1982 (1982-10-14), pages 905-907, XP000577331 ISSN: 0013-5194 page 906, colonne de gauche, ligne 3 - ligne 61	1,11

1

Formulaire PCT/ISA/210 (suite de la deuxième feuille) (juillet 1992)

RAPPORT DE RECHERCHE INTERNATIONALE

Renseignements relatifs aux membres de familles de brevets

Demande internationale No
PCT/FR 00/02715

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
EP 0381523 A	08-08-1990	JP 2204768 A	14-08-1990
		JP 3053367 A	07-03-1991
		US 5046094 A	03-09-1991
		JP 3073990 A	28-03-1991
		JP 3072737 A	27-03-1991
EP 0311470 A	12-04-1989	FR 2620248 A	10-03-1989
		AT 83573 T	15-01-1993
		AU 2197188 A	23-03-1989
		CA 1295706 A	11-02-1992
		DE 3876741 A	28-01-1993
		DE 3876741 T	24-06-1993
		ES 2037260 T	16-06-1993
		FI 884082 A,B,	08-03-1989
		JP 1133092 A	25-05-1989
		KR 9608209 B	20-06-1996
		US 5218637 A	08-06-1993
		US 5140634 A	18-08-1992

INTERNATIONAL SEARCH REPORT

International Application No
PCT/FR 00/02715

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L9/32

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data, PAJ, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 381 523 A (TOKYO SHIBAURA ELECTRIC CO) 8 August 1990 (1990-08-08) page 2, line 25 -page 3, line 7 ---	1,3,4
A	EP 0 311 470 A (TELEDIFFUSION FSE ;FRANCE ETAT (FR); PHILIPS NV (NL)) 12 April 1989 (1989-04-12) cited in the application abstract column 2, line 40 -column 3, line 50 column 12, line 30 -column 13, line 55 --- -/--	1,5,11

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
14 December 2000	29/12/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Masche, C

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Interna l Application No
PCT/FR 00/02715

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	QUISQUATER J -J ET AL: "FAST DECIPHERMENT ALGORITHM FOR RSA PUBLIC-KEY CRYPTOSYSTEM" ELECTRONICS LETTERS,IEE STEVENAGE,GB, vol. 18, no. 21, 14 October 1982 (1982-10-14), pages 905-907, XP000577331 ISSN: 0013-5194 page 906, left-hand column, line 3 - line 61 -----	1,11

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

Interna l Application No
PCT/FR 00/02715

Patent document cited in search report		Publication date	Patent family member(s)			Publication date
EP 0381523	A	08-08-1990	JP	2204768	A	14-08-1990
			JP	3053367	A	07-03-1991
			US	5046094	A	03-09-1991
			JP	3073990	A	28-03-1991
			JP	3072737	A	27-03-1991
EP 0311470	A	12-04-1989	FR	2620248	A	10-03-1989
			AT	83573	T	15-01-1993
			AU	2197188	A	23-03-1989
			CA	1295706	A	11-02-1992
			DE	3876741	A	28-01-1993
			DE	3876741	T	24-06-1993
			ES	2037260	T	16-06-1993
			FI	884082	A,B,	08-03-1989
			JP	1133092	A	25-05-1989
			KR	9608209	B	20-06-1996
			US	5218637	A	08-06-1993
			US	5140634	A	18-08-1992